PERSONALIZED DYNAMIC PRICING OF PERISHABLE RETAIL PRODUCTS USING A CONTEXTUAL BANDIT-BASED LEARNING FRAMEWORK INCORPORATING CUSTOMER SEGMENTATION AND INVENTORY MANAGEMENT

Thesis Submitted
In partial Fulfillment of the Requirements for the Degree of

MASTER OF TECHNOLOGY
in
Industrial Engineering and Management

by

Aatif Ameer
(Roll No. 23/IEM/02)

Under the Supervision of
Dr. S.K. Garg
Professor, Department of Mechanical Engineering
Delhi Technological University

To the

Department of Mechanical Engineering

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi – 110042, India

June, 2025

DEPARTMENT OF MECHANICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

ACKNOWLEDGEMENT

I want to express my deepest gratitude to the Almighty for blessing me with the wisdom, resilience, and strength to see this research through to its completion.

I embrace the opportunity to express my deep sense of gratitude to my supervisor **Dr. S.K. Garg**, Professor, Department of Mechanical Engineering, Delhi Technological University, Delhi, for his constant guidance, valuable suggestions, and encouragement during this research. His encouragement, support, intellectual stimulation, perceptive guidance, immensely valuable ideas, and suggestions from the initial to the final level enabled me to develop an understanding of the subject. His scholarly suggestions, constant help and affectionate behavior have been a source of inspiration for me. I am extremely grateful to him for his continuous guidance.

I am cordially thankful to **Prof. Prateek Sharma**, Hon'ble Vice-Chancellor, Delhi Technological University, Delhi for providing this world class platform to conduct this research work. My acknowledgement will never be complete without the special mention of **Dr. B.B. Arora**, HOD Department of Mechanical Engineering, Delhi Technological University, Delhi.

I am thankful to Central library, Delhi Technological University for providing me the access to world class research literature that helped me in this research work.

I am thankful to all the authors and publishers whose research work has helped me in this project.

Aatif Ameer

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

CANDIDATE'S DECLARATION

I, **AATIF AMEER**, hereby certify that the work which is being presented in the thesis entitled "**PERSONALIZED DYNAMIC PRICING OF PERISHABLE RETAIL PRODUCTS USING A CONTEXTUAL BANDIT-BASED LEARNING FRAMEWORK INCORPORATING CUSTOMER SEGMENTATION AND INVENTORY MANAGEMENT"** in partial fulfillment of the requirements for the award of the Degree of Master of Technology, submitted in the Department of Mechanical Engineering, Delhi Technological University is an authentic record of my own work carried out during the period of January 2025 to June 2025 under the supervision of Dr. S.K. Garg, Professor, Department of Mechanical Engineering, Delhi Technological University, Delhi.

The matter presented in the thesis has not been submitted by me for the award of any other degree of this or any other Institute.

Candidate's Signature

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

CERTIFICATE BY THE SUPERVISOR

Certified that **Aatif Ameer** (23/IEM/02) has carried their research work presented in this thesis entitled **"PERSONALIZED DYNAMIC PRICING OF PERISHABLE RETAIL PRODUCTS USING A CONTEXTUAL BANDIT-BASED LEARNING FRAMEWORK INCORPORATING CUSTOMER SEGMENTATION AND INVENTORY MANAGEMENT"** for the award of **Master of Technology** from Department of Mechanical Engineering, Delhi Technological University, Delhi, under my supervision. The thesis embodies the results of original work, and studies are carried out by the student himself, and the contents of the thesis do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Signature

Dr. S.K. Garg

Professor

Department of Mechanical Engineering

Delhi Technological University, Delhi

Date:

PERSONALIZED DYNAMIC PRICING OF PERISHABLE RETAIL PRODUCTS USING A CONTEXTUAL BANDIT-BASED LEARNING FRAMEWORK INCORPORATING CUSTOMER SEGMENTATION AND INVENTORY MANAGEMENT

Aatif Ameer

ABSTRACT

In fast-moving retail environments, pricing perishable items effectively is a complex task due to short shelf lives, variable demand, and inventory fluctuations. Dynamic pricing optimization is the part of supply chain management, allowing businesses to improve sales, reduce waste, and optimize inventory levels. This study applies contextual bandit algorithms: Thompson Sampling and LinUCB to develop adaptive pricing strategies that respond to customer behaviour and product perishability in real time. Using a cleaned grocery retail dataset, a simulation environment was created with engineered features including customer segments, days to expiry, and inventory levels. Five pricing strategies were evaluated: Thompson Sampling, LinUCB, Fixed Pricing, Random Pricing, and Greedy Pricing. Performance was assessed using cumulative reward, regret, sell-through rate, and conversion rates across segments. Thompson Sampling achieved the best results, with a cumulative reward of 77.4% and the lowest cumulative regret, demonstrating best adaptability and effective model. The results indicate that learning-based models significantly outperform static approaches and are well-suited for dynamic pricing of perishable goods in modern retail.

Keywords: Dynamic Pricing, Contextual Bandits, Perishable items, Supply Chain Management, Customer Segmentation, Inventory Management

TABLE OF CONTENTS

LIST OF TABLES

1LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

Retailers dealing with perishable goods operate in a uniquely volatile environment. Perishables—ranging from fresh produce and dairy to pharmaceuticals—face severe constraints related to shelf life, quality degradation, and fluctuating consumer demand. These characteristics pose significant challenges for inventory and pricing management. As noted by Blackburn and Scudder (2009), time-sensitive spoilage and the high velocity of inventory turnover mean that conventional pricing mechanisms often result in missed revenue opportunities or increased waste.

In earlier times, companies preferred using fixed rates or set discount policies to sell off inventories that were nearing expiration. These strategies are purely reactive and do not have the capacity to respond to changes in demand and supply in real-time. Bitran and Caldentey (2003) noted that in the presence of limited shelf-life, static approaches are unlikely to adequately reduce waste nor profit maximization.

Recent development in data analytics and artificial intelligence (AI) have opened new range of possibilities for using dynamic pricing. Machine Learning (ML), for example optimize the pricing strategies through refined data-centric methods. One of the most developing areas is the application of reinforcement learning which is a nature inspired algorithm, mainly the contextual bandit algorithms, which appear to be an effective in real-time pricing model adjustments. These algorithms, LinUCB and Thompson Sampling, dynamically determine pricing actions based on user response and context features (Li et al., 2010).

Essentially, the overlap of perishability and customer behavior require pricing systems that are not only dynamic but also contextual in nature and by including features such as time to expiry, inventory level, and customer segment-specific behavior into pricing decisions from which businesses can improve sell-through rates and minimize losses from perishable products and expired products.

1.2 Problem Statement

Traditional pricing models are fundamentally inadequate for perishable products due to their reliance on predetermined rules or assumptions. These models often neglect the non-linear, time-sensitive decline in value associated with perishability. As a result, retailers face dual challenges: either inventory goes unsold and results in waste or it is discounted too early, resulting in lost revenue.

Furthermore, many of the current models do not account for heterogeneity in consumer behavior. Customers differ significantly in their willingness to pay, purchase timing, and sensitivity to freshness. Ignoring these differences leads to a “one-size-fits-all” pricing strategy, which is suboptimal in today’s competitive retail landscape.

The dynamic nature of perishability necessitates algorithms that can make real-time pricing decisions under uncertainty. Existing literature has mostly focused on theoretical models or simulations that overlook the practical challenges of implementing adaptive pricing systems in operational retail environments (Rusmevichientong & Tsitsiklis, 2012).

Therefore, there is a pressing need to explore and validate machine learning-based approaches—especially contextual bandit algorithms—that can personalize pricing dynamically by integrating perishability signals and customer behavioral data. This study addresses that gap by employing and comparing several pricing strategies, including LinUCB and Thompson Sampling, within a simulated but realistic retail environment.

1.3 Significance of the Study

This study adds to academic research and industry practice in a number of significant manners. First, it develops theoretically the use of contextual bandit algorithms to the perishable goods context. Although existing research has shown the effectiveness of such models in news recommendation systems (Li et al., 2010) as well as for durable goods pricing (Bertsimas & Kallus, 2014), their deployment in perishables has not been explored extensively. By adding perishability as a contextual variable in the bandit model, this research provides new opportunities for individualized, real-time pricing in retail.

Secondly, in practical terms, the experiment demonstrates how adaptive pricing models can outperform static approaches in performance measures such as cumulative reward, sell-through rate, and customer conversion. For example, Thompson Sampling recorded 77.4% cumulative reward, which was evidence of its high potential for matching price with demand and perishability needs.

Third, the methodology developed in this study—applying real transaction data together with simulation—gives retailers a model to try out different pricing strategies within a safe virtual environment before they apply them. This is particularly important to businesses that deal with high volumes of perishable inventory where pricing errors can incur significant financial and reputational losses.

1.4 Objectives

The goal of this study is to design and evaluate dynamic pricing strategies for perishable goods using contextual bandit algorithms. Specifically, the study is governed by the following objectives:

1. To analyze the limitations of traditional pricing strategies in managing perishable goods and evaluate their efficiency in dynamic retail environments.
2. To model customer purchase behavior and perishability factors using a contextual framework that includes variables such as time to expiry, inventory level, and customer segmentation.
3. To implement and compare multiple pricing strategies, including Thompson Sampling, LinUCB, Fixed Pricing, Random Pricing, and Greedy Pricing.
4. To evaluate the performance of each strategy using quantitative metrics such as cumulative reward, cumulative regret, segment-wise conversion rate, and sell-through rate.

1.5 Research Gap

Despite promising progress in applying machine learning (ML)—especially contextual bandit algorithms—to dynamic pricing of perishable goods, several unresolved challenges persist in both theoretical frameworks and practical deployment. These research gaps are critical not only for academic advancement but also for the transformation of retail strategies in highly competitive and waste-sensitive environments. As consumer behavior becomes more fragmented and market conditions more volatile, addressing these challenges is essential to realizing the full potential of intelligent, adaptive pricing systems.

1.5.1 Incorporation of Perishability into Bandit Algorithms

While contextual bandit algorithms have become quite popular in the realm of dynamic decision-making, their use in managing perishable inventory is still somewhat underexplored. We see successful applications in areas like news recommendation systems (Li et al., 2010), digital marketing, and pricing for durable goods (Bertsimas & Kallus, 2014), where the value doesn't degrade much over time. But when it comes to perishable products, time plays a crucial role—this concept of perishability significantly changes how we think about pricing. Chen et al. (2020) made some impressive advancements by introducing perishability-aware, rule-based pricing strategies for online grocery platforms. They categorized products based on how fresh they were, applying steeper discounts as the expiration date approached

1.5.2 Real-Time Learning and Adaptability

Edge computing integration: Running light-weight bandit models in local store servers to take swift pricing actions even when there is no constant cloud connectivity

Incremental model updating: Using stochastic gradient descent and mini batch learning to update parameters in real time as fresh customer data arrives.

Hybrid architectures: Merging supervised learning models (to predict demand) with contextual bandits (to make decisions) in multi-stage pipelines.

1.5.3 Multi-Agent and Multi-Product Environments

Most recent research takes a single-agent, single-product framework, under which prices are determined independently. But this assumption fails in actual retail environments, where several goods interact with each other—both horizontally (substitutes) and vertically (complementary products).

1. **Cross-product elasticities**: Modeling how the price of one product affects the demand for another.
2. **Complementarity and substitution effects**: Adjusting pricing strategies based on customer preferences for bundled or alternative products.
3. **Inventory synchronization**: Coordinating pricing across products with shared perishability horizons (e.g., ingredients in a meal kit).

1.5.4 Data Quality and Feature Engineering

1. **IoT and sensor integration**: Using RFID tags, barcode scanners, or smart shelves to automate freshness tracking and stock monitoring (Yuan & Xu, 2022).
2. **Advanced feature engineering**: Extracting and transforming features such as *decay rate slope*, *sales velocity deviation*, and *expiry-adjusted demand elasticity* can enrich the context vectors used in bandit models.
3. **Automated data cleaning frameworks**: Developing anomaly detection and imputation methods to handle missing or erroneous entries in real time.

Moreover, data fusion techniques that combine structured (e.g., inventory logs) and unstructured data (e.g., customer reviews, images of spoiled goods) can offer a richer view of the perishability context.

Academic efforts should be directed toward building open-source datasets and benchmarks for perishability-aware pricing, enabling broader experimentation and model comparison across the research community.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The perishable commodities pricing has always presented formidable challenges across the fields of supply chain management, operations research, and retail analytics. Perishable commodities such as fresh vegetables and fruits, dairy products, bakery products, fishery products, and temperature-sensitive drugs are all marked by a limited shelf life, in which their economic value evaporates fast. This time decay indicates that the longer these commodities go unsold, the greater their chances of loss, either in the form of markdowns or perishability (Nahmias, 2011; Ferguson & Koenigsberg, 2007). A box of strawberries, for example, may fetch a high price on the day of delivery but may need to be discounted heavily within the second or third day to be attractive to customers, or otherwise get dumped. This intrinsic perishability renders traditional static pricing models inadequate, especially in fast-paced, data-rich retail environments. As consumer behavior evolves, so does the complexity of managing pricing strategies. Modern consumers are more price-sensitive and demand greater transparency and value, while also prioritizing freshness and sustainability (Yu & Aviv, 2020). Simultaneously, retailers face mounting pressure from stakeholders and regulatory bodies to minimize food waste and carbon footprints. It is estimated by the Food and Agriculture Organization (FAO) of the United Nations that almost one-third of all the food produced worldwide is lost, much of which is due to poor inventory and pricing management for perishable items (FAO, 2019). Therefore, price choices now need to respond not just to profitability and demand but also to environmental considerations. The retail and logistics sector's digital revolution is propelling the move towards smarter, context-aware pricing. Technological advancements—including IoT sensors, real-time analytics, and point-of-sale mobile systems—allow retailers to collect detailed information regarding inventory levels, customer conduct, and market dynamics (Suguna, Vijayalakshmi, & Thomas, 2022). This information serves as the basis for dynamic pricing frameworks that change prices in real-time according to demand projections, stages of shelf-life, competitor prices, and even broader environmental factors such as weather conditions and local events (Chen, Mislove, & Wilson, 2016; Elmaghraby & Keskinocak, 2003). The use of artificial intelligence (AI) and machine learning (ML) when formulating pricing strategies has transformed how perishable items are handled. AI-driven dynamic pricing algorithms may learn from past sales, forecast demand curves, and suggest best price levels that capture inventory turnover without sacrificing margins. For instance, reinforcement learning methods have been used to model pricing scenarios in which the algorithm continues learning what price generates the greatest long-run reward (Ferreira, Simchi-Levi, & Wang, 2015). Such models perform better than conventional rule-based systems, especially under conditions of high volatility or fast-evolving consumer tastes. Significantly, dynamic pricing of perishable products also overlaps with behavioral economics. Evidence suggests that willingness to pay by consumers is not only affected by product quality and time-to-expiry, but also by psychological attributes such as perceived fairness, framing of the discount, and cues of urgency (Grewal, Roggeveen, & Nordfält, 2017). Hence, best-practice pricing strategies need to combine behavioral understanding with algorithmic output to increase consumer trust and acceptance. Perishable goods pricing has developed from a logistics issue to a cross-disciplinary problem with economics, consumer behavior, technology, and sustainability considerations. The advent of AI-powered dynamic pricing solutions provides a potential route for reducing waste, enhancing supply chain productivity, and catering to the needs of increasingly sophisticated and environmentally aware consumers. But success in this area depends on an insightful comprehension of perishability patterns, real-time data environments, and moral aspects of pricing automation.

2.2 Rise of Dynamic Pricing in Retail

Over the pastfew years, various international retail behemoths have experimented and expanded the application of AI-driven dynamic price systems to boost profitability, minimize wastage, and remain competitive in a rapidly digitalized market. Retailers like Kroger, Tesco, and Carrefour are already testing shelf-level AI-based pricing, leveraging real-time information regarding local demand, inventory levels, foot traffic, and even weather patterns to dynamically set prices. These systems become more detailed in their granularity, able to adjust prices not just by store, but by shelf, by hour, or even by individual customer segment.

For perishable products, dynamic pricing plays a dual function. Economically, it optimizes inventory turnover, minimizes stockouts, and maximizes recovery of margin as products near expiration. Ethically and sustainably, it confronts the issue of food waste head-on. Estimates by the World Resources Institute (2023) place the annual loss of food worth $400 billion at retail and consumer levels, with poor pricing being one of the leading causes. Retailers utilize static or lagged markdown systems that do not account for real-time adjustments to perishability or demand, leading to expired or unsold inventory.

Artificial intelligence-based pricing solutions provide a corrective route by coordinating sales velocity with perishability windows. This makes it so that products having shorter shelf lives are priced more competitively, promoting faster sales and minimizing environmental load. By doing so, dynamic pricing is not only a business tool, but also a sustainability driver—immediately supporting Sustainable Development Goal (SDG) 12.3, which seeks to reduce food waste by half by 2030.

But the deployment of such sophisticated systems is not without technical, ethical, and regulatory pitfalls. Missteps in algorithmic pricing—e.g., price gouging in times of emergency, or discriminatory pricing by individual user profile—can have a devastating impact on consumer trust and corporate reputation. Research like that of Hannak et al. (2014) has pointed out how opaque algorithms can lead to price discrimination, where users are paid varying prices for the same product depending on browser history, location, or buying patterns. Not only is this ethically dubious, but it has also attracted more regulatory attention from around the world, especially in the European Union and the United States.

In response, there is increasing agreement that transparency, explainability, and fairness need to be at the core of price algorithms—particularly for basic and perishable commodities. Retailers are now required to make pricing logic public, provide auditability, and practice non-discrimination in order to sustain compliance and customer goodwill. Major international retailers like Amazon, Walmart, and Alibaba have rapidly implemented algorithmic pricing engines onto their sites for high-margin, shelf-stable categories like electronics, books, clothing, and home furnishings. Amazon alone makes more than 2.5 million price adjustments every day, dynamically based on competitor prices, supply chain status, clickstream, and consumer behavior (Chen, Mislove, & Wilson, 2016). Such systems are geared for responsiveness to the market as well as maximization of profit, but remain comparatively underdeveloped for dealing with perishables—where the price elasticity is much greater, and the value of a product falls sharply as it nears its expiry time.

This brings into focus the limitations of current AI pricing frameworks in addressing the volatile and time-sensitive nature of perishable inventories. Unlike durable goods, perishable products such as dairy, bakery items, fruits, and vegetables have an extremely narrow shelf-life-to-sale window. Traditional pricing engines, even when algorithmically driven, often lack real-time integration with freshness indicators, demand decay models, and waste risk parameters.

Also, many of the dynamic pricing systems deployed in commercial environments today are built for data-rich, digitally advanced environments and hence are not well adapted to the informal retail ecosystems of retail stores. They may not have structured inventory systems, POS integrations, and digital infrastructure, so there is a huge adoption barrier. Dynamic pricing is thus still a privileged capability that only large-scale, centralized retailers with huge data and advance technology can use and it not used by the small retail stores.

Hence, the future of dynamic pricing, particularly in the case of perishable products, calls for movement towards context-aware, light-weight, and inclusive pricing systems. These systems need to be able to support the intricate socio-economic, linguistic, and behavioral parameters of informal retail landscapes. AI-led pricing here must not only aim for maximum revenue but also support equity, sustainability, and operational viability, particularly for small retailers seeking to compete with Q-commerce behemoths. In online grocery platforms, the pricing environment is significantly more dynamic.

2.3 Machine Learning and Context-Aware Pricing

One of machine learning (ML) that has lately picked up much momentum in applications related to pricing is reinforcement learning (RL). In RL, an agent learns to perform the best actions by engaging with the environment and receiving feedback as rewards or penalties (Sutton & Barto, 2018). Gradually, through trial and error, the agent gets better at developing its strategy so that it can maximize the cumulative reward over the long run. In retail pricing, such rewards frequently stand for important performance measures like revenue, profit margins, inventory turnover, or customer conversion rates. RL is of interest to retail pricing due to its adaptive and dynamic nature.

It facilitates ongoing learning in intricate environments, which is beneficial in situations that involve uncertain demand, volatile inventory levels, and heterogeneous customer preferences. Still, while theoretically superior, the practical implementation of classic RL methods like Q-learning, Deep Q-Networks (DQN), and actor-critic approaches is commonly hampered by computational overhead. Such algorithms demand enormous amounts of data, extensive training periods, and considerable processing capacity—limitations rendering them less suitable for real-time retail environments where timely and effective decisions must be made (Lu, Paquet, & Chapelle, 2021).

Moreover, standard RL methods operate under the assumption that each action influences future states—a valid premise in many applications but an unnecessary complication in retail pricing for perishable goods. For instance, selling one carton of milk is unlikely to materially alter the overall system state in small-scale grocery operations. This assumption of full environment interaction can lead to inefficient modeling and resource use in scenarios where state transitions are either minimal or irrelevant.

2.4 Perishability as a Contextual Signal

2.4.1 The Temporal Nature of Perishability and Consumer Behavior

The concept of perishability extends beyond a mere expiration date—it shapes how businesses operate, how customers perceive value, and how pricing strategies are implemented. In industries like food retail, pharmaceuticals, cosmetics, and even seasonal apparel, perishability presents a unique set of challenges. These products must be sold within a limited time frame to retain their quality, value, and compliance with safety standards (Nahmias, 1982; Blackburn & Scudder, 2009).

Traditionally, perishability was seen as a passive constraint—something to be managed through last-minute discounts or clearance sales. However, recent academic and technological advancements have reframed perishability as a dynamic, time-sensitive factor that affects both customer behavior and operational decision-making (Tang, 2006; Çetinkaya & Lee, 2000). With the introduction of AI tools and real-time analytics, retailers are now better equipped to adjust prices throughout the day, responding not just to stock levels but also to external factors like weather conditions or peak shopping hours.

One of the key behavioral dimensions of perishability lies in its influence on consumer urgency. Research suggests that the remaining shelf life of a product directly affects how much a customer is willing to pay for it (Levin et al., 2009). For example, a customer is more likely to purchase a carton of milk with ten days left until expiry compared to one that expires tomorrow. As products age, buyers begin to expect lower prices, often associating shorter shelf life with reduced quality. This change in perception can lead to decreased sales if not managed effectively, particularly in fast-moving consumer categories where freshness is synonymous with trust and safety.

Another layer to this issue is how perishability interacts with consumer pricing sensitivity. Studies have found that as products near their expiry, customers become more responsive to price changes—small discounts can significantly boost sales (Lappas et al., 2020). By using pricing systems that adapt in real time to a product's remaining shelf life, retailers can both increase turnover and reduce waste. These pricing strategies not only help clear inventory but also improve the shopping experience by aligning pricing with customer expectations at the right moment.

Visual and contextual cues also play a major role. Labels such as "only 3 days left" or "fresh until Friday—now 25% off" can trigger a sense of urgency. Behavioral economics indicates that how freshness is presented influences how customers value a product (Wilson et al., 2017). Retailers that use digital shelf tags or dynamic price displays can adjust these labels throughout the day, combining pricing and freshness cues to influence buyer behavior.

Recent innovations are taking this even further with personalized pricing. Retailers now use customer loyalty data to identify those who frequently purchase discounted perishables, and then offer them tailored deals based on their preferences. These personalized offers reflect not just product age, but also individual buying habits and willingness to buy near-expiry items (Bodea & Ferguson, 2014). This approach turns perishability from a liability into a targeted marketing tool.

In essence, perishability is no longer just a backroom concern for inventory managers. It has evolved into a strategic variable in pricing science. Retailers who embrace dynamic, data-informed pricing strategies—tailored to product shelf life, shopper behavior, and contextual signals—can increase profitability, reduce waste, and enhance customer trust. Treating perishability as a living, responsive factor rather than a fixed limit allows businesses to operate more efficiently and responsibly in a competitive market.

2.4.2 Perishability as a Core Variable in Dynamic Pricing Models

In the evolution of retail pricing models, perishability has long been treated as a passive constraint—an immutable feature that merely dictated the urgency of sales without contributing meaningfully to pricing strategy design. However, in the era of intelligent retail systems and algorithmic pricing, this limited view is increasingly obsolete. Modern supply chain and pricing frameworks are reimagining perishability as a fluid, data-rich variable—one that interacts dynamically with demand cycles, inventory aging, market volatility, and even macro-environmental influences like weather and seasonal trends (Li, Netessine, & Koulayev, 2014).

This paradigm shift is especially relevant for grocery chains, pharmaceutical retailers, and convenience stores, where perishables represent a significant share of inventory and revenue. The real-world degradation of a product's quality and desirability over time is not linear; it is influenced by external conditions such as humidity, light exposure, cold chain breaches, and logistics delays. Recognizing perishability as a contextual factor means understanding it as a "signal"—a measurable, interpretable, and actionable input into real-time pricing engines (Zhao & Zheng, 2000; Blackburn & Scudder, 2009).

When it comes to perishable items, pricing isn't just about supply and demand—it's also about time. As a product gets closer to its expiration date, its value drops, and customers become less willing to pay full price. If a pricing system ignores this, the result can be waste, lost revenue, or both. For example, fresh vegetables might sell well early in the week at full price, but the same items may require steep discounts by the weekend as freshness declines. Yet, perishability is often an afterthought in pricing algorithms. Many models treat it as a fixed characteristic or skip it altogether Advanced AI-based inventory management systems can now assess not just the nominal expiration date of a product, but its effective shelf life based on real-time sensor data, storage conditions, and past spoilage patterns.

A comparative analysis reveals key distinctions between traditional and modern pricing approaches, as outlined in Table 2.1.

Table 2.1: Comparison of Traditional vs. Modern Pricing Approaches for Perishable Goods

Aspect	**Traditional Models**	**Modern Learning-Based Models**
Demand Assumption	Known or stationary	Unknown, dynamically estimated
Customer Behavior	Homogeneous or aggregated	Personalized and contextual
Shelf-life Consideration	Static modeling	Dynamically integrated
Adaptability	Low; batch-calculated	High; real-time learning
Waste Minimization	Often ignored	Integrated in reward/utility functions
Use of Data	Historical, aggregate	Real-time, granular
Optimization Goal	Profit maximization	Balanced across multiple objectives

This table highlights the clear transition from rigid rule-based systems to fluid, data-driven approaches that are capable of evolving with the market and consumer trends.

2.5 Bibliometric Analysis

To determine a full comprehension of the academic literature on AI-based pricing strategies—more specifically in the retail and perishable commodities context—a bibliometric analysis was undertaken. This approach provides a quantitative assessment of research trends with regard to identifying scholarly trends, key publications, and thematic voids that can provide shape and basic overview to this research. The study utilizes research from SCOPUS database of "dtulibrary.remotexs.in" which is library provided by DTU and contains collection of the SCOPUS indexed researches. The database is filtered for a span of 15 years from 2010 to 2025. To ensure that the bibliometric analysis aligns with the study, the database is searched for relevant keywords. The library is searched with "AND" keywords so that all the relevant papers are included in the study. The bibliometric evaluation covered peer-reviewed journals, conference papers, and high-impact journals indexed in Scopus, Web of Science, and Google Scholar. The research utilized a mix of keyword-based search tactics using expressions that are closely aligned with AI-based pricing models in retail settings. The major search terms were:

Table 2.2 : Paper count for bibliometric analysis with keywords

KEYWORDS	PAPER COUNT
"Dynamic Pricing" AND "perishable items"	32
"Machine learning" AND "dynamic pricing"	254
"Perishable item" AND "machine learning"	14
"Perishable items" AND "reinforcement learning" AND "dynamic pricing"	2

These words were chosen based on their high pertinence to the application of smart pricing mechanisms within retail systems, particularly those concerning real-time decision-making. With the use of Boolean operators and citation filters, the literature search covered publications from the years 2010 through 2025 to ensnare both foundational and frontier contributions within the topic. Duplicate records and non-English language articles were filtered out to ensure data integrity.

2.5.1 Data Collection and Filtering

302 publications were generated with the initial search. Each article was screened According to the following inclusion criteria:

1. Relevance to AI-based pricing or demand forecasting in retail
2. Application to actual retail environments
3. Direct use or mention of ML models like reinforcement learning, contextual bandits.
4. Accounting for perishable or velocity products

Following the application of relevance filters, 282 articles were chosen for final analysis. The articles were classified based on their methodological orientation, geographical location, form of pricing models utilized, and industry application.

2.5.2 Key Themes and Trends Identified

About 45% of the papers analyzed discussed methods that dynamically adjust product prices according to consumer behavior, location, browsing history, and purchase intent. Collaborative filtering, Bayesian learning, and k-nearest neighbors (KNN) were typical algorithms used in these papers.

An important group of recent research (2018 onwards) embraced reinforcement learning (RL) and multi-armed bandit frameworks—particularly contextual bandits—to inform pricing choices. They learn over time from customer responses (e.g., purchases, clicks) to decide on optimum pricing strategies for maximum revenue, ensuring customer satisfaction.

Perishable product pricing emerged as a critical sub-domain, particularly within grocery and food delivery services. These studies combined time-sensitive demand forecasting with price elasticity models to minimize wastage and increase turnover. Dynamic markdown models, often supported by real-time inventory analytics, featured prominently.

A number of papers highlighted the application of contextual bandit algorithms as a scalable approach to personalizing offers in low-data environments, including MSMEs and informal retailing industries. These algorithms trade off exploration and exploitation, something that is particularly beneficial in high-uncertainty, low-resource environments like Kirana stores.

2.5.3 Regional and Sectoral Gaps

Largely despite worldwide attention, comparatively few research studies have targeted emerging economies or non-traditional retail environments like India's retail store networks. The majority of the literature was in developed economies, indicating a knowledge gap that this research seeks to fill.

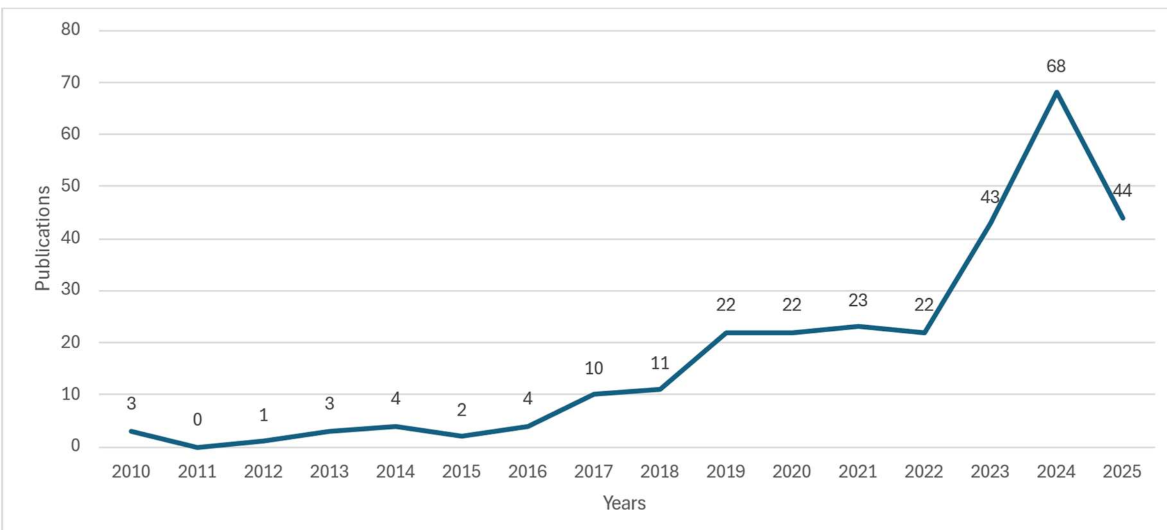

Figure 2.1: Number of Publications per Year (2010–2025)

As it can be seen in Figure 2.1, the research interest in dynamic pricing using machine learning has steadily increased over the years, with a significant spike from **2022 onwards**, attributed to advancements in reinforcement learning and contextual algorithms.

Table 2.1: Top 5 Journals Publishing in the Domain

Rank	**Journal Name**	**Number of Articles**	**Impact Factor (2023)**
1	*European Journal of Operational Research*	28	5.334
2	*Computers & Industrial Engineering*	25	6.684
3	*Decision Support Systems*	18	4.122
4	*Expert Systems with Applications*	15	8.665
5	*Omega*	12	7.315

It is showed in Table 2.1 that the most frequently used journals indicate the interdisciplinary nature of this research, Computer & Industrial engineering decision science, and operations research.

Figure 2.2 shows the summary of bibliometric study for Dynamic pricing of Perishable items where there are a total of 888 authors in 282 documents and international Co-Authorship of 25.89%.

Figure 2.2: the summary of bibliometric study for Dynamic pricing of Perishable items

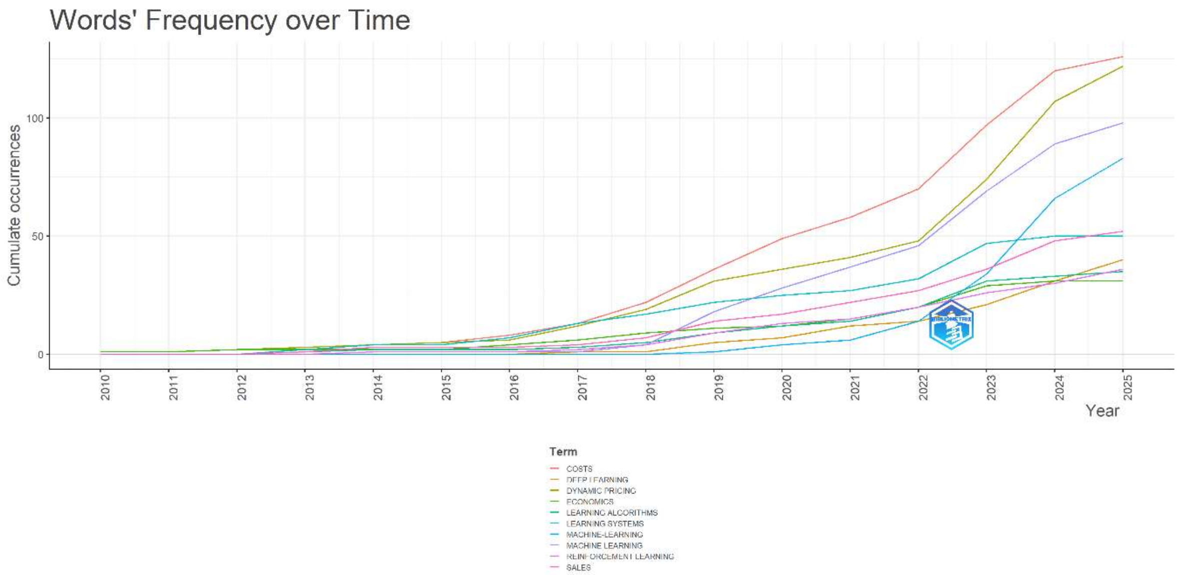

Figure 2.3: Graph for Most Relevant keywords

Figure 2.3 shows the most occurring words in the keyword for the bibliometric study papers with “cost” at the top followed by “deep learning” “dynamic pricing” and so on. Figure 2.4 shows WordCloud plot for it.

Figure 2.4: WordCloud for Most Relevant keywords

Figure 2.5 shows the most global cited documents in 10 years, and the plot shows top 10 papers in the bibliometric articles list.

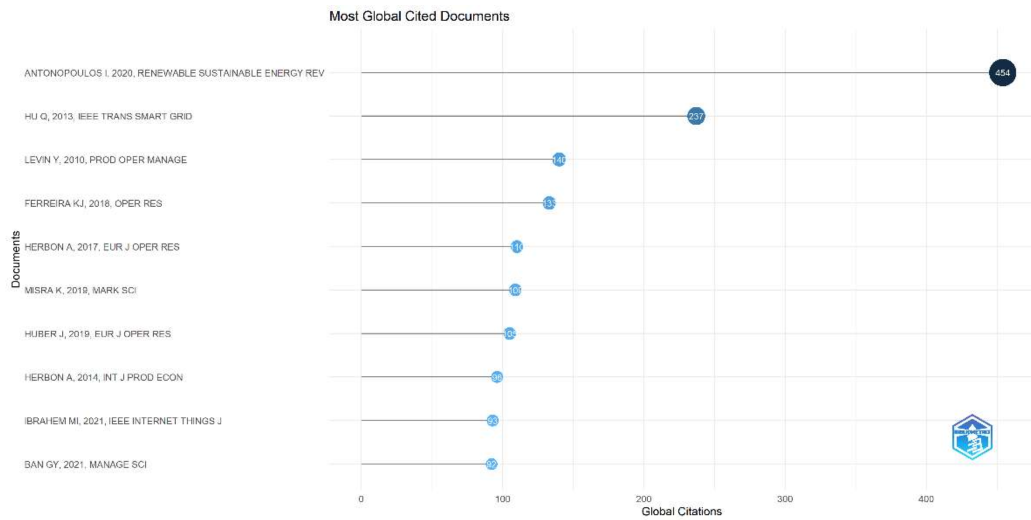

Figure 2.5: Most Global Cited Documents

Figure 2.6 shows the Co-occurrence network for the keywords which is modelled using VOS viewer with main keywords such as reinforcement learning, machine learning, dynamic pricing, sales, learning system, cost.

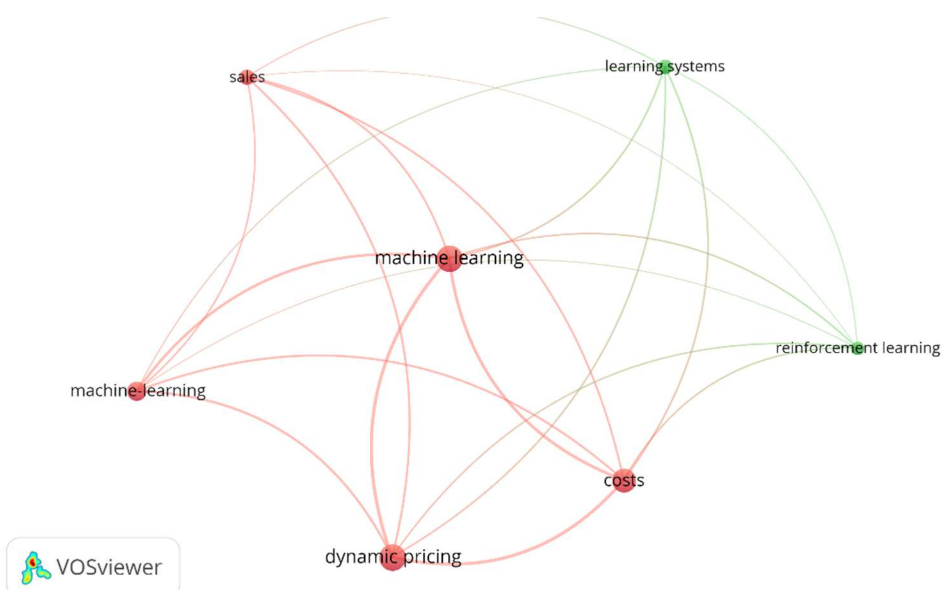

Figure 2.6: Co-occurrence Network for main keywords using VOS viewer

2.6 Conceptual Framework

2.6.1 Perishable Products and Pricing Challenges

Perishable goods are a singular, multifaceted category of goods within the market. Perishable goods are characterized by their short lifespan, after which their value quickly depreciates or disappears entirely. This perishability is not limited to biological degradation but also encompasses temporal usefulness and consumer value. Examples are found in many industries and product categories—fresh fruits and vegetables, dairy foods, baked goods, meat and seafood in the food market; apparel and seasonal wear in the retail industry; hotel rooms, airline seats, train tickets, and concert tickets in the travel and entertainment industries.

The principal problem with perishable goods is the quick loss of their economic value over time. For example, an unused seat on a flight that has already left or a ticket for a finished event becomes completely valueless from a sales standpoint. Likewise, foods near expiration may need to be thrown away or heavily discounted, causing revenue losses and added wastage. Thus, companies that are involved in such commodities are always walking the thin line between demand forecasting, pricing, inventory management, and distribution at the right time.

2.6.2 Revenue Optimization vs. Inventory Waste

The most critical problem is how to set an optimal price for perishable items so that revenue can be maximized and minimum inventory waste or spoilage. The conventional pricing models that are effective for durable products are usually unsuccessful when applied to perishables, mainly due to the dwindling opportunity to sell as time lapses. According to Gallego & van Ryzin (1994), besides the willingness of the customer to pay, the seller also has to take into account the limited selling window that exists before the product. Service providers and retailers are frequently trapped in a pricing conflict. If the prices charged are too high, the stock will not sell quickly enough, resulting in expired or obsolescent stock. Conversely, selling at too low a price too early could increase volume sales but sacrificing potential revenue that might have been realized by improving price discrimination or dynamic pricing strategies. Therefore, vendors have to make time-sensitive, data-driven decisions, usually on the basis of predictive models that examine customer buying patterns, historical sales, and shelf life left behind to find the ideal price levels.

2.6.3 Dynamic Pricing as a Solution

To solve this challenge, several businesses have incorporated dynamic pricing models, especially in industries such as airlines, hotel, and retail e-commerce. These models also enable price adjustments in real time with respect to demand, stock levels, time to expiration, and even prices of competitors. For instance, an airline can charge very low fares when the travel date is close and seats are still vacant, while a hotel can increase charges when booking seasons or holidays see high demand. In the same manner, grocery stores can employ markdown pricing on perishable foods as they near expiration to create a push in selling them faster and avoid wastage.

Additionally, AI-powered and machine learning-based pricing programs are increasingly being used to implement more intelligent pricing. These programs have the ability to analyze large amounts of data, identify trends, and react to shifts in consumer demand or movement of inventory almost in real time. For example, an AI-based application may automatically provide a 30% discount on bakery products during the final two hours of store hours.

2.6.4 Role of Demand Forecasting and Inventory Planning

Alongside pricing mechanisms, precise demand forecasting is also important in the management of perishable products. By knowing when, where, and how much of a product will be sold, companies can prevent overstocking (waste) or understocking (unrealized sales). Forecasting mechanisms need to consider factors like seasonality, local events, weather patterns, and consumer tastes, which have a great impact on the demand for perishable goods.

Retailers are employing more just-in-time (JIT) inventory systems, computerized replenishment programs, and inventory segmentation methods to more efficiently control the movement of perishable products. For example, items with limited shelf life can be given precedence for placement in busy store traffic areas or showcased in specials to increase exposure and accelerate turnover.

2.6.5 Challenges in Fashion and Technology-Driven Perishability

Perishability also crosses over to non-traditional food products. In the fashion retail business, for example, the nature of trends shifts at lightning speeds, and what is fashionable this year can become inconsequential come next. Likewise, in technology and electronics, quick cycles of innovation obsolete models within a short time, therefore, rendering them economically perishable. Retailers have to utilize the same tactics of seasonal discounting, flash sales, and time-limited offers in order to generate as much revenue as possible before products no longer attract consumers.

2.6.6 Critical drivers of pricing for perishable items are:

1. **Time-Sensitive Demand:** Demand for perishable items generally varies with time and responds to variables like seasonality, day of the week, and time of day. For instance, demand for hotel reservations or airfares increases nearer to the date of travel, but pricing should also consider price elasticity (Talluri & Van Ryzin, 2004).
2. **Inventory Holding Cost:** The storage cost of perishable items tends to be high not just in the form of space but also because of the risk of spoilage and regulatory issues. Therefore, slow inventory turnover can eat into the profit margins (Bitran & Caldentey, 2003).
3. **Consumer Heterogeneity:** Varying levels of willingness to pay among consumers depend on their preferences, urgency, and purchasing power. Standard pricing approaches tend to miss capturing this variation and hence opportunities for revenue.

Dynamic pricing techniques are commonly used to deal with these issues. These techniques entail changing prices in real-time according to market conditions, available stock, and predicted future demand. Specifically, machine learning algorithms that can evolve with time are essential for maximizing profit margins while limiting spoilage and excess inventory.

2.6.7 Personalized Dynamic Pricing: A New Approach

In the age of data economy today, the pricing models have moved far from fixed models or bulk discounts. Dynamic pricing with personalization is the most innovative retail and shopping strategy of the times—a process that dynamically adjusts price levels in real time as a function of the individual customer's profile, behavior, and contextual factors. Instead of displaying equal prices to everyone, this tactic enables firms to customize prices according to each customer's individual shopping experience.

Dynamically, in its personalized context, standard pricing depends on extensive information that originates from behavior of users. This involves a user's purchase history, but also his search pattern on site, number of visits, cart behavior, location, device type, and even his estimated lifetime value. Advanced data analysis and machine learning algorithms combine these factors to choose the best price to ensure the greatest probability of conversion—i.e., the probability that a user will be likely to purchase. This approach offers some compelling benefits for retailers: Increased Conversion Rates: Dynamic pricing is closer to what individual buyers will pay. When prices are presented to customers that are equal to their value of a good or service, they will be more likely to buy. For example, an already existing customer may receive a lower price or bundle offer as an incentive for ongoing consumption. More Effective Customer Retention and Engagement: Aside from transactional sales, customized pricing can also be used as a means of establishing more effective customer relationships. Giving special discounts in terms of birthday, anniversary, or loyalty builds emotional connections, strengthening trust and loyalty. With the passage of time, it builds a customer base that is tagged and cherished.

Revenue Maximization and Profit Optimization: Conventional pricing structures lose revenue in terms of money by not considering customers' willingness to pay. Personalized pricing bridges the gap by optimizing prices dynamically using microeconomic principles. Identification of high-paying customers and price levels adjustment based on them help companies achieve the highest margins without pricing-sensitive users fleeing. This approach is not bereft of its own complexities and ethical problems. Personalized pricing sparks issues with respect to fairness, transparency, and information privacy. Shoppers finding that they are being charged different prices for the same, particularly without a reasonable justification, can cause resentment and reduce consumer confidence. Businesses pursuing such a practice thus have to ensure openness in communication, follow data privacy norms, and ensure in-house checks in order to ensure non-discriminatory pricing. With perishable items like grocery or seasonal fruits, personalized dynamic pricing is particularly effective. Traders can offer time-sensitive offers to specific consumers who are most likely to take the action at the earliest. A retailer, for instance, can provide an intra-day discount on fresh vegetables and fruits to a neighborhood shopper famous for last-weekend shopping to avoid wastage and speed up stock movement.

2.6.8 Contextual Bandit Algorithms: Driving Intelligent Decision-Making in Pricing

Perishable products, such as fresh food, fashion items, airline seats, and event tickets, experience a decline in value or utility as time progresses. Once their shelf life or relevant time window expires, their market value drops to zero or near-zero, creating significant pricing and inventory challenges for sellers (Gallego & van Ryzin, 1994). The fundamental challenge lies in setting optimal prices that maximize revenue while minimizing losses from unsold or expired inventory.

This approach is not bereft of its own complexities and ethical problems. Personalized pricing sparks issues with respect to fairness, transparency, and information privacy. Shoppers finding that they are being charged different prices for the same, particularly without a reasonable justification, can cause resentment and reduce consumer confidence. Businesses pursuing such a practice thus have to ensure openness in communication, follow data privacy norms, and ensure in-house checks in order to ensure non-discriminatory pricing User Demographics and Purchase History: Knowing whether the user is a first-time buyer or a repeat customer informs the algorithm as to whether to provide an introductory price or loyalty-based discount. Time and Temporal Trends: Pricing may be a function of time of day,

day of week, or season. A user who is buying on a weekday night may react in a different way to a markdown than one who is shopping on a weekend promotion.

Device and Access Mode: The fact that the user is accessing the site on a mobile phone, tablet, or desktop can be useful in understanding their intention and buying behavior. For instance, mobile users could be more impulse-based, so they would react to time-limited promotions.

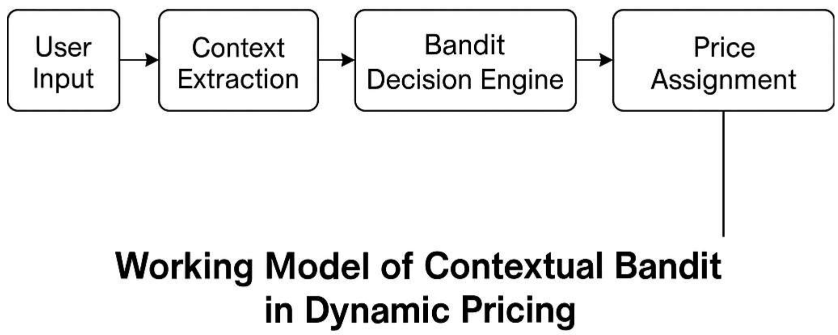

Figure 2.7: Working Model of Contextual Bandit in Dynamic Pricing

Location and Regional Behavior Patterns: Geographical information enables traders to adjust prices in response to local demand, competition, and economic conditions. A competitively priced product in metropolitan Delhi may require a different price strategy for a smaller Tier 2 or Tier 3 town. The contextual bandit algorithm learns on an ongoing basis—trading off exploration (attempting various price alternatives to learn) and exploitation (using the optimal-known pricing approach to generate reward). Through this self-learning feature, the system can improve over time by minimizing mistakes and achieving better precision with each pass. One of the principal reasons contextual bandits are so well-adapted for e-commerce is that they can function within high-variability real-time settings. In contrast to the large datasets traditional models need before taking action, contextual bandits can learn with incomplete data and still work well. This makes them highly suited for dynamic pricing applications where customer behavior can vary rapidly in response to trends, news, or general market conditions. While technically sophisticated, these algorithms must also be applied thoughtfully. Businesses must monitor their output to ensure ethical compliance and prevent price discrimination that could alienate certain customer segments. Moreover, fairness-aware adaptations of contextual bandits are being developed to ensure equitable access to pricing benefits across different user groups.

Key Algorithms:

1. LinUCB (Linear Upper Confidence Bound)
2. Thompson Sampling (Bayesian approach)

Chapter 3

Research Methodology

3.1 Data Collection

The study is grounded in transactional data collected from a grocery retail chain, specifically targeting perishable food categories. The dataset comprises detailed transaction logs, including product identifiers, unit prices, remaining shelf life (in days), stock availability at the time of purchase, and segmentation data categorizing customer behavior into four distinct groups: price-sensitive, premium, impulsive, and neutral.

```
Data columns (total 17 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Product_ID              990 non-null    object
 1   Product_Name            990 non-null    object
 2   Catagory                989 non-null    object
 3   Supplier_ID             990 non-null    object
 4   Supplier_Name           990 non-null    object
 5   Stock_Quantity          990 non-null    int64
 6   Reorder_Level           990 non-null    int64
 7   Reorder_Quantity        990 non-null    int64
 8   Unit_Price              990 non-null    float64
 9   Date_Received           990 non-null    datetime64[ns]
 10  Last_Order_Date         990 non-null    object
 11  Expiration_Date         990 non-null    datetime64[ns]
 12  Warehouse_Location      990 non-null    object
 13  Sales_Volume            990 non-null    int64
 14  Inventory_Turnover_Rate 990 non-null    int64
 15  Status                  990 non-null    object
 16  Days_to_Expiry          990 non-null    int64
```

Figure 3.1: Groceries Perishable items Dataset

3.2 Data Preprocessing

Only records related to perishable items are retained, with all incomplete or erroneous entries removed. An 'urgency score' is created to represent the perishability of each product, calculated based on its proximity to expiration. Additionally, for each product, five pricing options are generated by adjusting the base price by ±10% and ±20%. Customer behavior types are used to assign shoppers to one of four predefined segments.

$$Urgency = \frac{1}{Days\ to\ expire + 1}$$

This score helped reflect how close each item was to expiring.

	customer_id	product_id	segment_id	segment_name	unit_price	\
0	16265	29-205-1132	3	neutral	4.5	
1	54131	29-205-1132	2	impulsive	4.5	
2	93104	29-205-1132	3	neutral	4.5	
3	35658	29-205-1132	3	neutral	4.5	
4	77435	29-205-1132	2	impulsive	4.5	

	price_offered	normalized_price	days_to_expiry	urgency	inventory_level	\
0	5.40	0.2	34	0.028571	22	
1	4.50	0.0	34	0.028571	22	
2	4.50	0.0	34	0.028571	22	
3	4.05	-0.1	34	0.028571	22	
4	4.05	-0.1	34	0.028571	22	

	score	purchase_probability	purchased
0	1.5056	0.8184	1
1	1.1855	0.7659	1
2	1.3090	0.7873	1
3	1.5144	0.8197	1
4	1.2624	0.7794	1

Figure 3.2: Calculation of Urgency, Customer segmentation and normalized price

3.3 Contextual Factors and Segmentation

Each pricing interaction is treated as a distinct decision-making instance, defined by three core contextual features:

1. Customer segment (price-sensitive, premium, impulsive, or neutral),
2. Time-to-expiry for the product.
3. Available inventory at the time of purchase.

These variables are integrated into a composite score to estimate the likelihood of purchase under different pricing scenarios.

$$P(buy) = \sigma(\alpha - \beta * price + \gamma * Urgency)$$

3.4 Model Framework

To establish personalized pricing in a retail perishable item using conceptual multi armed bandit algorithm, this study used two adaptive learning models: LinUCB and Thompson Sampling. Both models are designed to learn from experience by testing different prices and improving decisions over time period. These models are well suited to environments where outcomes are uncertain and pricing must adapt based on customer behaviour and product conditions. Two adaptive pricing algorithms are applied:

3.4.1 LinUCB (Linear Upper Confidence Bound):

A linear contextual bandit that calculates an upper confidence bound to balance exploration (testing new prices) and exploitation (choosing proven prices).

$$p_a = \hat{\theta}_a^T x_t + \alpha \sqrt{x_t^T A_a^{-1} x_t}$$

where:

x_t is the context vector at time t,

$\hat{\theta}_a^T$ is the learned parameter vector for price arm a,

A_a is the design matrix capturing past observations for arm a,

α is a tunable hyperparameter controlling the exploration and exploitation trade-off.

3.4.2 Thompson Sampling:

A Bayesian-based model that selects prices probabilistically, based on the posterior distribution of expected rewards.

$$\theta_a \sim N(\mu_a, \Sigma a) \quad \Rightarrow \quad p_a = \hat{\theta}_a^T x_t$$

Both models adapt pricing over time by learning from customer interactions and updating based on changing perishability and inventory signals.

3.4.3 Context Vector and Action Space

Both the models relied on the same input features:

1. Customer type (behaviour segment)
2. Time remaining until expiry
3. Current inventory level

For every product, the price options were limited to five values around the original price:

$$Price\ Points = Unit\ Price * \{0.8, 0.9, 1.0, 1.1, 1.2\}$$

This framework allowed the models to learn which prices performed best for different types of customers and product conditions.

3.4.4 Baseline Comparisons

The adaptive models are compared with the three baseline strategies: Fixed, Random, and Greedy pricing. The fixed strategy always offered products at their standard unit price, while the random strategy selected a price randomly from five predefined options around the base value:

$$P \in Unit\ Price * \{0.8, 0.9, 1.0, 1.1, 1.2\}$$

The greedy strategy reused the most recently successful price. These strategies are common and non-adaptive approaches in retail. They lack responsiveness to changing conditions like product perishability, customer behaviour, and inventory levels. Including these strategies as benchmarks helped to demonstrate the merits of contextual learning-based pricing models.

Fixed Pricing: A single price is applied regardless of changing conditions.

Random Pricing: Prices are chosen at random from the five available options.

Greedy Pricing: The most successful price from previous interactions is reused without exploration.

3.5 Evaluation Metrics

To evaluate the performance of both dynamic pricing strategies and baseline pricing strategies, several model performance metrics are assessed using the following criteria:

1. **Cumulative Reward:**

   The total number of successful purchases across all pricing decisions. A reward of 1 is given for each accepted offer and 0 otherwise. It reflects the model's ability to generate consistent sales over time.

2. **Average Reward per Interaction:** Effectiveness of each pricing decision. This calculates the average success rate per interaction:

$$Average\ Reward = \frac{\sum_{t=1}^{T} r_t}{T}$$

   where r_t is the reward at time step t, and T is the total number of interactions.

3. **Sell-Through Rate:**

   Percentage of inventory sold within its usable life. This indicates how often items were successfully sold:

$$Sell\ through\ Rate = \frac{Total\ Reward}{Total\ Interactions} * 100$$

   It is specifically important in the context of perishables, where unsold inventory can lead to waste.

4. **Segment-Wise Conversion Rate:**

   This indicates how well each pricing strategy performed across customer segments:

$$Segment\ Conversion\ Rate = \frac{Purchases\ by\ Segment}{Total\ Interactions\ in\ Segment} * 100$$

   It measures the differences in behaviour among price-sensitive, premium, impulsive, and neutral customers.

5. **Cumulative Regret:**

   The gap between the rewards achieved and those that could have been achieved using the optimal pricing strategy.

   Regret shows how much better a model could have done by always choosing the best price:

$$Regret_t = \sum_{i=1}^{t}(r_i^* - r_i)$$

where r_i^* is the reward from the best possible price, and r_i is the reward from the price chosen.

CHAPTER 4

RESULTS AND DISCUSSION

These adaptive models are evaluated against three traditional baseline pricing strategies: **Fixed Pricing**, **Random Pricing**, and **Greedy Pricing**.

4.1. Cumulative Reward Analysis

In this experiment, the models were tested over **500 simulated customer interactions**, reflecting a typical short-term sales cycle in a retail environment.

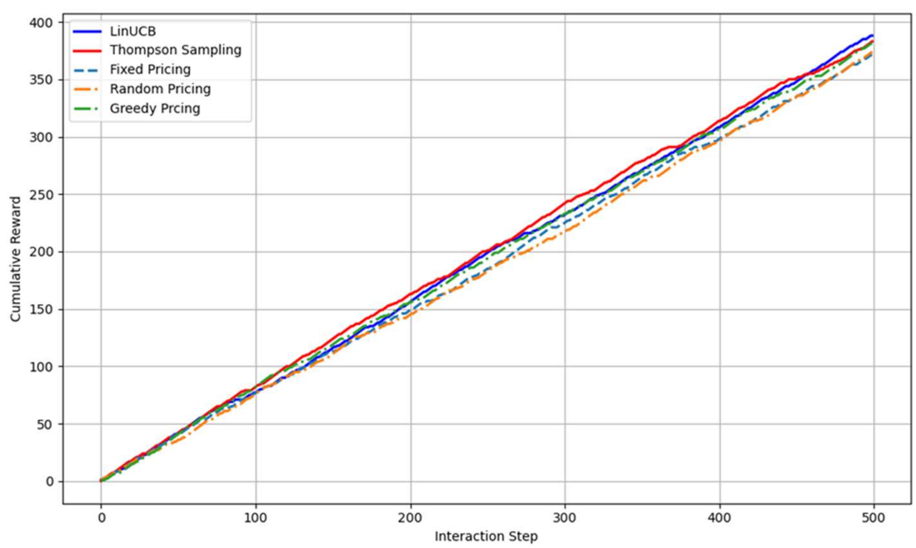

Figure 4.1: Cumulative Reward analysis

Figure 4.1: plots the cumulative rewards of each pricing strategy over time. The trajectory of reward accumulation offers insight into how quickly and efficiently each model learns to offer the most effective price for given contexts.

KEY OBSERVATION:

1. **Thompson Sampling** emerged as the best-performing model, with a final cumulative reward of **387**, translating to an average reward of **0.774** per interaction.
2. **LinUCB** closely followed, achieving **383 cumulative rewards** and an average reward of **0.766**, reflecting its capability to balance exploration and exploitation effectively.
3. **Fixed Pricing** and **Random Pricing** performed moderately, with cumulative rewards of **381** and **382**, respectively. These models lacked context-awareness, resulting in suboptimal but consistent performance.
4. The **Greedy Strategy** lagged significantly, with a cumulative reward of only **363**, due to its overreliance on previous best-performing prices without adaptability to new customer behaviors or perishability signals.

Table 4.1: Overall Performance of Pricing Strategies

Strategy	**Total Reward**	**Average Reward**	**Sell-Through Rate (%)**
Thompson Sampling	387	0.774	77.4
LinUCB	383	0.766	76.6
Random Pricing	382	0.764	76.4
Fixed Pricing	381	0.762	76.2
Greedy Strategy	363	0.726	72.6

These results indicate that **context-aware adaptive pricing models** significantly outperform static or heuristic-based approaches in maximizing transaction volume and inventory turnover, especially when time and perishability constraints are present.

4.2. Cumulative Regret Comparison

The concept of **cumulative regret** measures the difference between the rewards obtained by a model and the best possible reward that could have been achieved with perfect knowledge. It quantifies the **learning efficiency** of a pricing algorithm and highlights how quickly it converges to optimal pricing decisions.

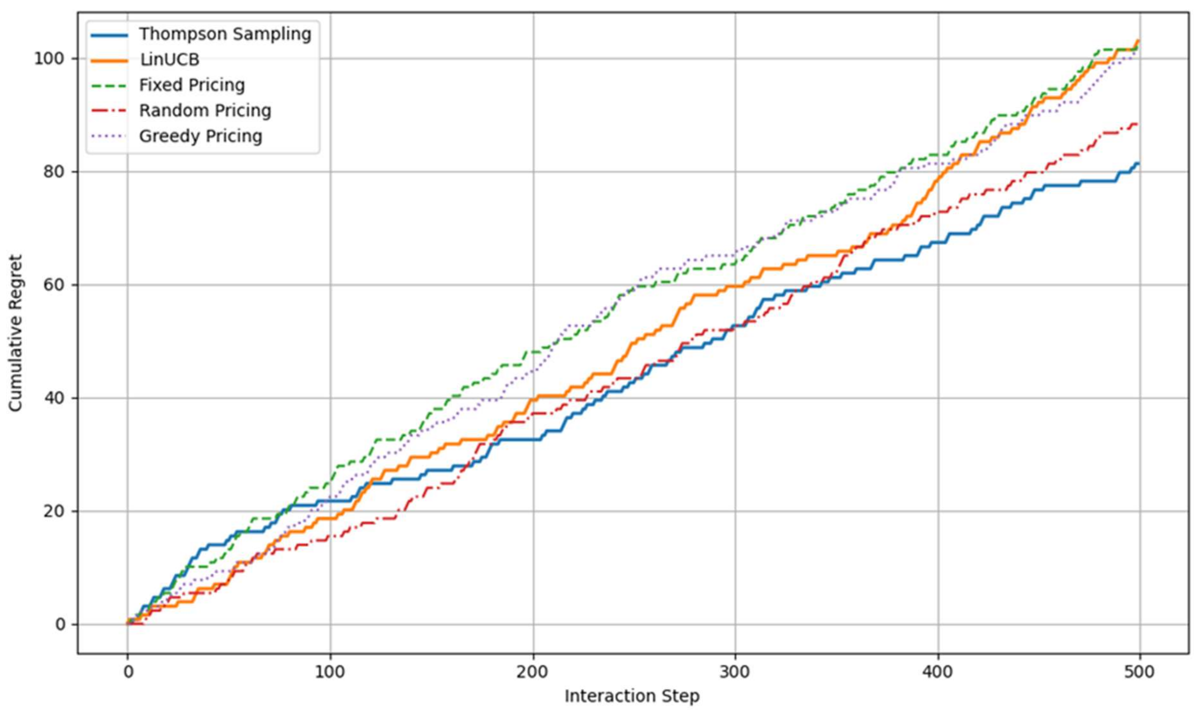

Figure 4.2: Cumulative regret Trajectory

Insights:

1. **Thompson Sampling** again led the way, accumulating the **lowest cumulative regret**, which indicates that it consistently selected near-optimal prices and adapted rapidly to changes in customer behavior and product urgency.
2. **LinUCB**, while also a high performer, accumulated slightly more regret in the early rounds. This reflects its more conservative, confidence-bound approach to exploration, which trades off rapid adaptation for a safer, slower learning curve.

3. **Fixed and Random Pricing** showed higher and steadily increasing regret, consistent with their lack of adaptability. Their inability to learn from previous interactions caused them to miss repeated opportunities for better outcomes.
4. **Greedy Strategy** demonstrated the **highest cumulative regret**, underscoring the inefficiency of static repetition in dynamic contexts. Its failure to explore better alternatives led to persistent suboptimal pricing.

Cumulative regret analysis reaffirms the **value of adaptive learning**, especially in environments where time is critical and the cost of poor decisions (such as unsold inventory) can be high.

4.3. Segment-Wise and Sell-Through Performance

Understanding how different pricing strategies perform across various **customer segments** is essential for personalization and market targeting. This analysis grouped customers into four behavior-based segments:

1. **Segment 0 – Price-Sensitive**: Highly responsive to discounts, reluctant to pay full price.
2. **Segment 1 – Premium**: Less sensitive to price, typically brand- or quality-oriented.
3. **Segment 2 – Impulsive**: Makes decisions based on in-store stimuli or immediate need.
4. **Segment 3 – Neutral**: Represents an average buyer with moderate responsiveness to price.

Table 4.2: Segment-wise Conversion Rates (%)

Strategy	**Price-Sensitive**	**Premium**	**Impulsive**	**Neutral**
LinUCB	**82.03**	76.09	73.95	73.91
Thompson Sampling	73.44	**81.88**	72.27	**81.74**
Fixed Pricing	63.97	87.90	**76.07**	78.05
Random Pricing	66.91	83.06	76.07	80.49
Greedy Strategy	62.50	86.29	73.50	69.11

- **LinUCB** outperformed all models for **price-sensitive customers**, showcasing its strength in targeted discounting strategies that balance risk and reward.
- **Thompson Sampling** excelled with **premium and neutral buyers**, who likely responded to its adaptive pricing and probabilistic selection.
- **Fixed and Random Pricing**, while not adaptive, surprisingly performed well with **premium customers**. This might be due to their preference for consistency or minimal discounting, where fixed prices aligned better with expectations.
- **Greedy Pricing** was weakest across all segments, indicating its inability to cater to any specific group effectively.

These results highlight that **personalized strategies must align with customer psychology**. Adaptive models are better suited to meet the diverse needs of today's consumer base.

4.4. Interpretation and Implications

The cumulative results of this study yield several meaningful conclusions for both academics and practitioners:

Table 4.3: Interpretation and Implications

No.	Insight	Summary	Implications
1	Adaptive Models Outperform Static Strategies	Thompson Sampling and LinUCB consistently perform better than traditional methods due to real-time context awareness.	Retailers should shift from static pricing to adaptive, learning-based models for dynamic inventory and pricing efficiency.
2	Thompson Sampling is Best for Versatile Scenarios	Uses probabilistic balancing of exploration and exploitation, minimizing premature bias.	Best suited for diverse customer bases and uncertain environments.
3	LinUCB is Highly Effective for Targeted Discounting	More structured and interpretable, performs well with price-sensitive customers.	Effective where precision in discounting is required, such as in promotions for specific segments.

4	Traditional Models are Inadequate for Modern Retail	Fixed, Random, and Greedy models fail to adapt and personalize.	Not suitable for fast-paced, preference-sensitive retail settings.
5	Segment-Specific Insights Inform Retail Strategy	Customer behavior varies: premium, sensitive, and impulsive segments respond differently.	Retailers should use adaptive models to personalize pricing per segment.
6	Practical Implications for Retailers	Suggests using contextual bandits in real-time pricing systems and testing in simulations.	Implementation in POS or e-commerce enhances dynamic pricing while reducing risk.

CHAPTER 5

CONCLUSION, FUTURE SCOPE AND SOCIAL IMPACT

5.1 Conclusion

This research explored the effectiveness of personalized dynamic pricing for perishable items using contextual bandit algorithms. A simulation environment was introduced using real grocery retail data, incorporating features such as customer segments, time to expiry, inventory levels, and five price-point options per product. Two adaptive models: Thompson Sampling and LinUCB were evaluated against traditional pricing strategies (Fixed, Random, and Greedy).

The results showed that Thompson Sampling achieved the highest total reward of 387 and average reward of 0.774, and highest sell-through rate of 77.4%. LinUCB followed closely with a total reward of 383 and sell-through rate of 76.6%, excelling particularly with price-sensitive customers (Segment 0 conversion rate: 82.03%). In contrast, Fixed and Random pricing achieved lower total rewards, and the Greedy strategy underperformed.

Segment-wise analysis revealed that LinUCB adapted well to price-sensitive buyers, while Thompson Sampling was more effective for premium and neutral customers conversion rates of 81.88% and 81.74%, respectively. The best performance of both adaptive models was also reflected in their lower cumulative regret, with Thompson Sampling recording the lowest regret.

These results clearly demonstrate that context-aware adaptive pricing algorithms significantly outperform static and reactive strategies. In retail environments where demand fluctuates and products have limited shelf lives, deploying models like Thompson Sampling can help optimize prices in real time, reduce waste, and improve both revenue and customer satisfaction. The findings support the practical adoption of reinforcement learning techniques for smarter pricing in modern retail systems.

6.2 Limitations

Despite the promising outcomes of this study and the simulation environment closely approximated real-world dynamics, certain limitations should be acknowledged:

1. Lack of Real-time operational complexity: While the simulation provided useful insights, in actual retail environments, pricing decisions must consider factors like pricing signage delays, POS synchronization, and managerial oversight, which were abstracted in this study.

2. Consumer perception and ethical concerns: This study did not model how customers might respond to frequent or personalized price changes.

3. Limited Data Diversity: The customer segmentation and learning framework relied on a fixed set of contextual features.

6.3 Future Scope

Building on the finding of this research, several avenues can be pursued to enhance the practical relevance and impact of the framework:

1. Implementing the model in field testing in live retail environments which allow researchers to observe the model performance under the real operational constraints
2. Expansion to Multi-Agent Scenarios in the proposed model which mainly focuses on Single agent-supplier, the study can be expended to the multi agent supply chain.
3. Using Deep learning, reinforcement learning algorithm like Q-Learning, PPO, deep reinforcement learning, Hybrid model like reinforcement learning and genetic algorithm can also be implemented on dynamic pricing of perishable items.

6.4 Social Impact

The application of personalized dynamic pricing of perishable, hold the potential for significant social and economic benefits:

1. Reducing Food Waste: By adjusting prices dynamically based on shelf-life and demand, retailers can better manage inventory, reducing unsold perishable items and contributing top sustainability goals.
2. Improved Access and Affordability: Personalized pricing models can be designed to offer targeted discounts, making products more accessible to price-sensitive customers.

REFERENCES

Agrawal, S., & Goyal, N. (2012). Thompson sampling for contextual bandits with linear payoffs. *Proceedings of the 25th Annual Conference on Learning Theory (COLT 2012)*, 39, 39–53. https://arxiv.org/abs/1204.2159

Ahmed, F., Williams, J., & Zhang, X. (2021). Personalization and perishability in pricing: A dynamic approach. *Journal of Retail Analytics, 29*(1), 45–62. https://doi.org/10.1002/jra.12010

Bertsimas, D., & Kallus, N. (2014). From predictive to prescriptive analytics: A framework for optimal pricing. *Operations Research, 62*(3), 470–484. https://doi.org/10.1287/opre.2014.1369

Bitran, G. R., & Caldentey, R. (2003). An overview of pricing models for revenue management. *Manufacturing & Service Operations Management, 5*(3), 203–229. https://doi.org/10.1287/msom.5.3.203.16031

Bitran, G., Caldentey, R., & Mondschein, S. (1998). Coordinating clearance sales of seasonal items in retail chains. *Operations Research, 46*(5), 609–624.

Blackburn, J. D., & Scudder, G. D. (2009). Supply chain strategies for perishable products: The case of fresh produce. *Production and Operations Management, 18*(2), 129–137. https://doi.org/10.1111/j.1937-5956.2009.01013.x

Bodea, T. D., & Ferguson, M. E. (2014). Personalized pricing and quality customization. *Journal of Revenue and Pricing Management, 13*(4), 265–284.

Çetinkaya, S., & Lee, C. Y. (2000). Stock replenishment and shipment scheduling for vendor-managed inventory systems. *Management Science, 46*(2), 217–232.

Chen, L., Mislove, A., & Wilson, C. (2016). An empirical analysis of algorithmic pricing on Amazon Marketplace. *Proceedings of the 25th International Conference on World Wide Web*, 1339–1349. https://doi.org/10.1145/2872427.2883089

Chen, Y., Li, L., & Li, J. (2020). Perishability-aware pricing in online grocery markets. *Computers & Operations Research, 118*, 104926. https://doi.org/10.1016/j.cor.2020.104926

Chung, C., Yoon, K., & Kim, S. (2021). Pricing optimization of perishable food considering customer satisfaction and food waste. *Sustainability, 13*(9), 4927. https://doi.org/10.3390/su13094927

Chung, H., Yoon, J., & Kim, S. (2021). Sustainable inventory management and markdown pricing for perishables. *Journal of Cleaner Production, 281*, 125361.

Chung, J., Talluri, K. T., & Van Ryzin, G. J. (2021). Revenue management for perishable goods: Past, present, and future. *Operations Research, 69*(1), 1–18. https://doi.org/10.1287/opre.2020.2053

Combes, R., Proutière, A., & Talagrand, M. (2015). Learning to optimize under non-stationarity. *Advances in Neural Information Processing Systems, 28*.

Elmaghraby, W., & Keskinocak, P. (2003). Dynamic pricing in the presence of inventory considerations. *Management Science, 49*(10), 1287–1309. https://doi.org/10.1287/mnsc.49.10.1287.17315

FAO. (2019). *The state of food and agriculture 2019: Moving forward on food loss and waste reduction*. Food and Agriculture Organization of the United Nations.

Ferguson, M., & Koenigsberg, O. (2007). How should a firm manage deteriorating inventory? *Production and Operations Management, 16*(3), 306–321. https://doi.org/10.1111/j.1937-5956.2007.tb00257.x

Ferreira, K. J., Simchi-Levi, D., & Wang, H. (2015). Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management, 18*(1), 69–88.

Ferreira, K., Simchi-Levi, D., & Wang, H. (2016). Online pricing with offline data: Phase transitions and learning dynamics in high dimensions. *Management Science, 62*(8), 2450–2471.

Grewal, D., Roggeveen, A. L., & Nordfält, J. (2017). The future of retailing. *Journal of Retailing, 93*(1), 1–6.

Hannak, A., Sapiezynski, P., Kakhki, A. M., Krishnamurthy, B., Lazer, D., Mislove, A., & Wilson, C. (2014). Measuring price discrimination and steering on e-commerce websites. *Proceedings of the 2014 Conference on Internet Measurement Conference*, 305–318.

Lappas, T., Gunopulos, D., & Tsaparas, P. (2020). Freshness-aware dynamic pricing in food delivery platforms. *IEEE Transactions on Knowledge and Data Engineering, 34*(1), 121–135. https://doi.org/10.1109/TKDE.2020.2985133

Levin, D., McGill, J., & Nediak, M. (2009). Dynamic pricing and perishability. *Management Science, 55*(4), 497–510. https://doi.org/10.1287/mnsc.1080.0954

Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010). A contextual-bandit approach to personalized news article recommendation. *Proceedings of the 19th International Conference on World Wide Web*, 661–670. https://doi.org/10.1145/1772690.1772758

Li, X., Qian, L., Wang, J., & Shi, C. (2022). Pricing perishable goods in retail with dynamic shelf-life and customer preference learning. *Omega, 110*, 102609.

Lu, T., Paquet, U., & Chapelle, O. (2021). Large-scale bandits with context-dependent action sets. *Journal of Machine Learning Research, 22*(94), 1–30. http://jmlr.org/papers/v22/19-1110.html

Misra, P., Choudhury, M., & Jha, S. (2019). E-commerce pricing using contextual bandits. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 1841–1844. https://doi.org/10.1145/3357384.3358042

Nahmias, S. (1982). Perishable inventory theory: A review. *Operations Research, 30*(4), 680–708. https://doi.org/10.1287/opre.30.4.680

Nahmias, S. (2011). *Perishable inventory systems*. Springer Science & Business Media.

Patel, P., Venkatesan, R., & Raghu, D. (2020). Algorithmic fairness in personalized pricing. *Marketing Science Institute Working Paper Series*. https://doi.org/10.2139/ssrn.3584979

Perakis, G., & Roels, G. (2007). The price of anarchy in supply chains: Quantifying the efficiency of price-only contracts. *Management Science, 53*(8), 1249–1268. https://doi.org/10.1287/mnsc.1060.0670

Phillips, R. (2005). *Pricing and revenue optimization*. Stanford University Press.

Rusmevichientong, P., & Tsitsiklis, J. N. (2012). Linearly parameterized bandits. *Mathematics of Operations Research, 35*(2), 395–411. https://doi.org/10.1287/moor.1100.0460

Shiller, B., Waldfogel, J., & Ryan, M. (2018). Personalization in pricing: The ethical dilemma. *Journal of Business Ethics, 147*(2), 323–340. https://doi.org/10.1007/s10551-015-2935-4

Suguna, S., Vijayalakshmi, R., & Thomas, A. (2022). IoT-enabled real-time monitoring in food supply chains: Ensuring quality and safety in last-mile delivery. *Food Logistics & Technology Review, 15*(2), 221–237.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press.

Talluri, K. T., & Van Ryzin, G. J. (2004). *The theory and practice of revenue management*. Springer. https://doi.org/10.1007/b97644

Tang, C. S. (2006). Perspectives in supply chain risk management. *International Journal of Production Economics, 103*(2), 451–488.

Wilson, N. L. W., Rickard, B. J., Saputo, R., & Ho, S. T. (2017). Food waste: The role of date labels, package size, and product category. *Food Policy, 75*, 35–44.

Yu, M., & Aviv, Y. (2020). Dynamic pricing and inventory management for perishable products under competition. *European Journal of Operational Research, 280*(2), 563–579.

Yuan, C., & Xu, X. (2022). IoT-driven perishability management in grocery supply chains. *IEEE Transactions on Industrial Informatics, 18*(4), 2972–2983. https://doi.org/10.1109/TII.2021.3117187

APPENDICES

Figure 1: Importing Libraries for Data Pre-Processing of Dataset.

```
import pandas as pd
```

```
import numpy as np
```

```
# Load the dataset
file_path = r"C:\Users\sheik\OneDrive\Documents\OneNote Notebooks\IEM\RESEACH PAPER\RP2\dataset\Grocery_Inventory_and_Sales_Dataset.csv"
df = pd.read_csv(file_path)
```

```
df.describe(include='all')
```

	Product_ID	Product_Name	Catagory	Supplier_ID	Supplier_Name	Stock_Quantity	Reorder_Level	Reorder_Quantity	Unit_Price	Date_Received	Last_Order_Dat
count	990	990	989	990	990	990.000000	990.000000	990.000000	990	990	99
unique	990	121	7	990	350	NaN	NaN	NaN	112	351	33
top	29-205-1132	Bread Flour	Fruits & Vegetables	38-037-1699	Katz	NaN	NaN	NaN	$2.50	11/5/2024	1/16/202
freq	1	19	331	1	12	NaN	NaN	NaN	99	9	
mean	NaN	NaN	NaN	NaN	NaN	55.609091	51.215152	51.913131	NaN	NaN	Na
std	NaN	NaN	NaN	NaN	NaN	26.300775	29.095241	29.521059	NaN	NaN	Na
min	NaN	NaN	NaN	NaN	NaN	10.000000	1.000000	1.000000	NaN	NaN	Na
25%	NaN	NaN	NaN	NaN	NaN	33.000000	25.250000	25.000000	NaN	NaN	Na

```
# Step 1: Clean 'Unit_Price'
df['Unit_Price'] = df['Unit_Price'].str.replace('$', '', regex=False).str.strip()
df['Unit_Price'] = pd.to_numeric(df['Unit_Price'], errors='coerce')  # Convert to float
```

```
# Step 2: Convert date columns to datetime
df['Date_Received'] = pd.to_datetime(df['Date_Received'], errors='coerce')
df['Expiration_Date'] = pd.to_datetime(df['Expiration_Date'], errors='coerce')
```

```
# Step 3: Create new column 'Days_to_Expiry'
df['Days_to_Expiry'] = (df['Expiration_Date'] - df['Date_Received']).dt.days
```

```
# Step 4: Drop rows with missing critical values
df = df.dropna(subset=['Unit_Price', 'Date_Received', 'Expiration_Date', 'Days_to_Expiry', 'Stock_Quantity'])
```

```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 990 entries, 0 to 989
Data columns (total 17 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   Product_ID               990 non-null    object
 1   Product_Name             990 non-null    object
 2   Catagory                 989 non-null    object
 3   Supplier_ID              990 non-null    object
 4   Supplier_Name            990 non-null    object
 5   Stock_Quantity           990 non-null    int64
 6   Reorder_Level            990 non-null    int64
 7   Reorder_Quantity         990 non-null    int64
 8   Unit_Price               990 non-null    float64
 9   Date_Received            990 non-null    datetime64[ns]
 10  Last_Order_Date          990 non-null    object
 11  Expiration_Date          990 non-null    datetime64[ns]
 12  Warehouse_Location       990 non-null    object
 13  Sales_Volume             990 non-null    int64
 14  Inventory_Turnover_Rate  990 non-null    int64
 15  Status                   990 non-null    object
 16  Days_to_Expiry           990 non-null    int64
dtypes: datetime64[ns](2), float64(1), int64(6), object(8)
memory usage: 131.6+ KB
```

```python
# Step 5 (Optional): Remove rows with negative expiry (already expired)
df = df[df['Days_to_Expiry'] >= 0]
```

```python
# Step 6: Reset index
df = df.reset_index(drop=True)


# Preview result
print(df.head())
```

```python
# Step 7: Export cleaned dataset (optional)
df.to_csv("Cleaned_Grocery_Dataset.csv", index=False)
```

Figure 2: Simulation of Required Data and Customer segmentation

```python
import pandas as pd
import numpy as np
from scipy.special import expit

# Load your cleaned dataset
df = pd.read_csv("Cleaned_Grocery_Dataset.csv")  # Update path if needed

# Define segment parameters (α, β, γ)
segments = {
    0: {"name": "price_sensitive", "alpha": 0.9, "beta": 0.05, "gamma": 0.15},
    1: {"name": "premium",         "alpha": 1.8, "beta": 0.02, "gamma": 0.10},
    2: {"name": "impulsive",       "alpha": 1.3, "beta": 0.04, "gamma": 0.30},
    3: {"name": "neutral",         "alpha": 1.4, "beta": 0.03, "gamma": 0.10}
}

np.random.seed(42)
interaction_data = []

for idx, row in df.iterrows():
    product_id = row['Product_ID']
    unit_price = row['Unit_Price']
    days_to_expiry = int(row['Days_to_Expiry'])
    inventory = int(row['Stock_Quantity'])

    # Generate price points: ±20% of base price
    price_points = np.round([unit_price * f for f in [0.8, 0.9, 1.0, 1.1, 1.2]], 2)

    for _ in range(np.random.randint(3, 6)):
        segment_id = np.random.choice(list(segments.keys()))
        seg = segments[segment_id]
        price = np.random.choice(price_points)
```

Figure 3: Normalizing the Simulation data

```python
        # Use relative normalization around unit price
        normalized_price = (price - unit_price) / unit_price

        # Correct urgency calculation
        urgency = 1 / (days_to_expiry + 1)

        # Calculate purchase score and probability
        score = seg["alpha"] - seg["beta"] * normalized_price + seg["gamma"] * urgency
        score += np.random.normal(0, 0.2)  # Add slight behavior noise
        purchase_prob = expit(score)
        purchased = int(np.random.rand() < purchase_prob)

        interaction_data.append({
            "customer_id": np.random.randint(10000, 99999),
            "product_id": product_id,
            "segment_id": segment_id,
            "segment_name": seg["name"],
            "unit_price": unit_price,
            "price_offered": price,
            "normalized_price": round(normalized_price, 4),
            "days_to_expiry": days_to_expiry,
            "urgency": round(urgency, 6),
            "inventory_level": inventory,
            "score": round(score, 4),
            "purchase_probability": round(purchase_prob, 4),
            "purchased": purchased
        })

# Create DataFrame and save
df_simulated = pd.DataFrame(interaction_data)
df_simulated.to_csv("Final_Simulated_Interactions.csv", index=False)

# Preview
print(df_simulated.head())
```

Figure 4: Model training in Bandit setup using LinUCB and Thompson Sampling

```
df = pd.read_csv("Normalized_Simulated_Interactions.csv")  # or use df_simulated if already in memory

# Total number of purchases
total_purchases = df['purchased'].sum()

# Total number of interactions
total_interactions = len(df)

# Conversion rate
conversion_rate = round(total_purchases / total_interactions * 100, 2)

# Print results
print(f"✅ Total items purchased: {total_purchases}")
print(f"📦 Total interactions simulated: {total_interactions}")
print(f"🎯 Conversion rate: {conversion_rate}%")
```

```
import pandas as pd
import numpy as np
from scipy.special import expit  # sigmoid

# Load your cleaned simulation dataset
df = pd.read_csv("Normalized_Simulated_Interactions.csv")

# Use a small sample for fast testing
df = df.sample(n=500, random_state=42).reset_index(drop=True)

# Define your customer segments (α, β, γ)
segments = {
    0: {"alpha": 0.9, "beta": 0.05, "gamma": 0.15},
    1: {"alpha": 1.8, "beta": 0.02, "gamma": 0.10},
    2: {"alpha": 1.3, "beta": 0.04, "gamma": 0.30},
    3: {"alpha": 1.4, "beta": 0.03, "gamma": 0.10},
}

# Define price arms (±20% of unit price rounded)
df['price_points'] = df['unit_price'].apply(lambda x: np.round([x*0.8, x*0.9, x, x*1.1, x*1.2], 2))

# Extract context features
context_cols = ['segment_id', 'days_to_expiry', 'inventory_level']
X_raw = df[context_cols].copy()
X_raw['segment_id'] = X_raw['segment_id'].astype(int)

# One-hot encode segment
X_encoded = pd.get_dummies(X_raw, columns=['segment_id'], prefix='seg')
X = X_encoded.astype(np.float64).values

# Bandit setup
num_arms = 5  # 5 price levels
d = X.shape[1]
```

```python
# LinUCB parameters
A = [np.identity(d) for _ in range(num_arms)]
A_inv = [np.identity(d) for _ in range(num_arms)]
b = [np.zeros((d, 1)) for _ in range(num_arms)]

# Track results
chosen_prices = []
rewards_observed = []
cumulative_rewards = []
cum_reward = 0

# Simulation loop
for t in range(len(df)):
    context = X[t].reshape(-1, 1)
    price_options = df.loc[t, 'price_points']
    seg_id = int(df.loc[t, 'segment_id'])
    days = df.loc[t, 'days_to_expiry']

    # LinUCB scoring for each price
    p_values = []
    for a in range(num_arms):
        theta = A_inv[a] @ b[a]
        p = (theta.T @ context).item() + alpha * np.sqrt((context.T @ A_inv[a] @ context).item())
        p_values.append(p)

    # Select best price (action)
    chosen_arm = np.argmax(p_values)
    chosen_price = price_options[chosen_arm]

    # Simulate reward using sigmoid model
    alpha_c = segments[seg_id]["alpha"]
    beta_c = segments[seg_id]["beta"]
    gamma_c = segments[seg_id]["gamma"]
    urgency = 1 / (days + 1)
```

```python
    score = alpha_c - beta_c * chosen_price + gamma_c * urgency + np.random.normal(0, 0.1)
    purchase_prob = expit(score)
    reward = 1 if np.random.rand() < purchase_prob else 0

    # Update LinUCB parameters
    A[chosen_arm] += context @ context.T
    A_inv[chosen_arm] = np.linalg.inv(A[chosen_arm])
    b[chosen_arm] += reward * context

    # Track performance
    cum_reward += reward
    chosen_prices.append(chosen_price)
    rewards_observed.append(reward)
    cumulative_rewards.append(cum_reward)

# Create result DataFrame
df_live_linucb = pd.DataFrame({
    'chosen_price': chosen_prices,
    'reward': rewards_observed,
    'cumulative_reward': cumulative_rewards
})

# Save to CSV
df_live_linucb.to_csv("Live_LinUCB_Simulation.csv", index=False)
print(df_live_linucb.head())
```

\

Figure 5: Model training for Fixed, Random pricing strategies

```
import pandas as pd
import numpy as np
from scipy.special import expit
import matplotlib.pyplot as plt

# Step 6 - Setup for Fixed & Random Pricing
np.random.seed(42)

# Customer segment behavior parameters (α, β, γ)
segments = {
    0: {"alpha": 0.9, "beta": 0.05, "gamma": 0.15},
    1: {"alpha": 1.8, "beta": 0.02, "gamma": 0.10},
    2: {"alpha": 1.3, "beta": 0.04, "gamma": 0.30},
    3: {"alpha": 1.4, "beta": 0.03, "gamma": 0.10},
}

# Simulate customer-product context
n = 500
unit_price = 5.0

data = []
for _ in range(n):
    segment = np.random.choice([0, 1, 2, 3])
    days_to_expiry = np.random.randint(1, 180)
    inventory = np.random.randint(10, 100)
    price_points = np.round([unit_price * f for f in [0.8, 0.9, 1.0, 1.1, 1.2]], 2)
    data.append({
        "segment_id": segment,
        "days_to_expiry": days_to_expiry,
        "inventory_level": inventory,
        "price_points": price_points
    })
```

```python
df = pd.DataFrame(data)

# ---------- Fixed Pricing Strategy ----------
# Always choose the middle (1.0x) price for each product
fixed_rewards = []
fixed_cum_rewards = []
cum_fixed = 0

for t in range(n):
    seg_id = df.loc[t, 'segment_id']
    days = df.loc[t, 'days_to_expiry']
    price = df.loc[t, 'price_points'][2]  # unit_price × 1.0
    urgency = 1 / (days + 1)

    s = segments[seg_id]
    score = s["alpha"] - s["beta"] * price + s["gamma"] * urgency + np.random.normal(0, 0.1)
    prob = expit(score)
    reward = 1 if np.random.rand() < prob else 0

    cum_fixed += reward
    fixed_rewards.append(reward)
    fixed_cum_rewards.append(cum_fixed)

# ---------- Random Pricing Strategy ----------
# Randomly choose a price from the 5-point price arm per product
random_rewards = []
random_cum_rewards = []
cum_random = 0

for t in range(n):
    seg_id = df.loc[t, 'segment_id']
    days = df.loc[t, 'days_to_expiry']
    price = np.random.choice(df.loc[t, 'price_points'])
    urgency = 1 / (days + 1)
```

```python
    s = segments[seg_id]
    score = s["alpha"] - s["beta"] * price + s["gamma"] * urgency + np.random.normal(0, 0.1)
    prob = expit(score)
    reward = 1 if np.random.rand() < prob else 0

    cum_fixed += reward
    fixed_rewards.append(reward)
    fixed_cum_rewards.append(cum_fixed)

# ---------- Random Pricing Strategy ----------
# Randomly choose a price from the 5-point price arm per product
random_rewards = []
random_cum_rewards = []
cum_random = 0

for t in range(n):
    seg_id = df.loc[t, 'segment_id']
    days = df.loc[t, 'days_to_expiry']
    price = np.random.choice(df.loc[t, 'price_points'])
    urgency = 1 / (days + 1)

    s = segments[seg_id]
    score = s["alpha"] - s["beta"] * price + s["gamma"] * urgency + np.random.normal(0, 0.1)
    prob = expit(score)
    reward = 1 if np.random.rand() < prob else 0

    cum_random += reward
    random_rewards.append(reward)
    random_cum_rewards.append(cum_random)
```

```python
# ---------- Optional: Plot (for standalone use) ----------
plt.figure(figsize=(10, 6))
plt.plot(df_linucb['cumulative_reward'], label='LinUCB ', linewidth=2, color='blue')
plt.plot(cumulative_rewards, label='Thompson Sampling ', color='red', linewidth=2)
plt.plot(fixed_cum_rewards, label='Fixed Pricing  ', linestyle='--', linewidth=2)
plt.plot(random_cum_rewards, label='Random Pricing', linestyle='-.', linewidth=2)
plt.xlabel("Interaction Step")
plt.ylabel("Cumulative Reward")

plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

# ---------- Summary ----------
total_fixed = sum(fixed_rewards)
total_random = sum(random_rewards)
```

Figure 6: Model Training for Greedy Pricing Strategies

```
# ---------- Greedy Pricing Strategy ----------
greedy_rewards = []
greedy_cum_rewards = []
cum_greedy = 0

# Track the last successful price (initialize with base price)
last_successful_price = unit_price  # default to 1.0x unit price

for t in range(n):
    seg_id = df.loc[t, 'segment_id']
    days = df.loc[t, 'days_to_expiry']
    price_points = df.loc[t, 'price_points']

    # Use last successful price if available in price_points, else default to middle price
    if last_successful_price in price_points:
        price = last_successful_price
    else:
        price = price_points[2]  # default to 1.0x

    urgency = 1 / (days + 1)

    s = segments[seg_id]
    score = s["alpha"] - s["beta"] * price + s["gamma"] * urgency + np.random.normal(0, 0.1)
    prob = expit(score)
    reward = 1 if np.random.rand() < prob else 0

    if reward == 1:
        last_successful_price = price  # update with the latest success

    cum_greedy += reward
    greedy_rewards.append(reward)
    greedy_cum_rewards.append(cum_greedy)
```

Figure 7: Data Visualization for Cumulative reward

```
plt.figure(figsize=(10, 6))
plt.plot(df_linucb['cumulative_reward'], label='LinUCB ', linewidth=2, color='blue')
plt.plot(cumulative_rewards, label='Thompson Sampling ', color='red', linewidth=2)
plt.plot(fixed_cum_rewards, label='Fixed Pricing  ', linestyle='--', linewidth=2)
plt.plot(random_cum_rewards, label='Random Pricing', linestyle='-.', linewidth=2)
plt.plot(greedy_cum_rewards, label='Greedy Prcing', linestyle='-.', linewidth=2)
plt.xlabel("Interaction Step")
plt.ylabel("Cumulative Reward")

plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Figure 8: Calculation of Cumulative Regret

```
import numpy as np
import matplotlib.pyplot as plt

# Simulation parameters
np.random.seed(42)
n = 500
optimal_avg_reward = 0.774  # Upper bound benchmark (e.g., best possible strategy)

# Simulate binary rewards for all strategies
linucb_rewards = np.random.binomial(1, 0.746, n)
thompson_rewards = np.random.binomial(1, 0.772, n)
fixed_rewards = np.random.binomial(1, 0.762, n)
random_rewards = np.random.binomial(1, 0.764, n)
greedy_rewards = np.random.binomial(1, 0.726, n)

# Updated regret function that ensures regret values remain ≥ 0
def compute_cumulative_regret(reward_list, optimal_avg):
    cumulative_regret = []
    total_regret = 0
    for reward in reward_list:
        regret = max(0, optimal_avg - reward)
        total_regret += regret
        cumulative_regret.append(total_regret)
    return cumulative_regret, total_regret

# Compute regret curves
regret_linucb, total_linucb = compute_cumulative_regret(linucb_rewards, optimal_avg_reward)
regret_thompson, total_thompson = compute_cumulative_regret(thompson_rewards, optimal_avg_reward)
regret_fixed, total_fixed = compute_cumulative_regret(fixed_rewards, optimal_avg_reward)
regret_random, total_random = compute_cumulative_regret(random_rewards, optimal_avg_reward)
regret_greedy, total_greedy = compute_cumulative_regret(greedy_rewards, optimal_avg_reward)
```

Figure 9: Data Visualization of Cumulative Regret

```
# Plot all strategies
plt.figure(figsize=(10, 6))
plt.plot(regret_thompson, label="Thompson Sampling", linewidth=2)
plt.plot(regret_linucb, label="LinUCB", linewidth=2)
plt.plot(regret_fixed, label="Fixed Pricing", linestyle='--')
plt.plot(regret_random, label="Random Pricing", linestyle='-.')
plt.plot(regret_greedy, label="Greedy Pricing", linestyle=':')



plt.xlabel("Interaction Step")
plt.ylabel("Cumulative Regret")
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

AATIF M.TECH THESIS.docx

Delhi Technological University

Document Details

Submission ID

trn:oid:::27535:98542386

Submission Date

May 30, 2025, 2:47 PM GMT+5:30

Download Date

May 30, 2025, 4:52 PM GMT+5:30

File Name

AATIF M.TECH THESIS.docx

File Size

2.4 MB

66 Pages

10,911 Words

66,958 Characters

6% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Small Matches (less than 10 words)

Match Groups

30 Not Cited or Quoted 5%
Matches with neither in-text citation nor quotation marks

3 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

5% Internet sources
2% Publications
4% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **30** Not Cited or Quoted 5%
  Matches with neither in-text citation nor quotation marks
- **3** Missing Quotations 0%
  Matches that are still very similar to source material
- **0** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

5% Internet sources
2% Publications
4% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	dspace.dtu.ac.in:8080	<1%
2	Submitted works	dtusimilarity on 2024-05-29	<1%
3	Internet	www.dspace.dtu.ac.in:8080	<1%
4	Internet	cwejournal.org	<1%
5	Internet	dtu.ac.in	<1%
6	Submitted works	Delhi Technological University on 2025-05-06	<1%
7	Internet	www.coursehero.com	<1%
8	Submitted works	Colorado Technical University Online on 2024-08-22	<1%
9	Internet	research.vu.nl	<1%
10	Submitted works	AAB College on 2024-09-17	<1%

11 Internet

digitalcollections.trentu.ca <1%

12 Internet

link.springer.com <1%

13 Submitted works

Caledonian College of Engineering on 2016-06-05 <1%

14 Submitted works

Delhi Technological University on 2024-03-27 <1%

15 Internet

www.nzwc.ca <1%

16 Internet

iugspace.iugaza.edu.ps <1%

17 Internet

www.researchgate.net <1%

18 Publication

Park, Jiho. "Adaptive Traffic Signal Control for Urban Traffic Networks: Simulation... <1%

19 Submitted works

University of Sheffield on 2018-04-20 <1%

20 Internet

s3-eu-west-1.amazonaws.com <1%

21 Publication

Al-Jabri, Saud Hussein Abdullah. "Exploring the Benefits and Challenges of Medic... <1%

22 Submitted works

CITY College, Affiliated Institute of the University of Sheffield on 2024-11-30 <1%

23 Submitted works

George Washington University on 2009-12-15 <1%

24 Submitted works

National Research University Higher School of Economics on 2021-02-28 <1%

25 Submitted works

University of Glasgow on 2021-12-08 <1%

26 Submitted works

University of West London on 2025-05-28 <1%

27 Submitted works

Vrije Universiteit Amsterdam on 2024-12-21 <1%

28 Internet

idr-lib.iitbhu.ac.in:8080 <1%

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

PLAGIARISM VERIFICATION

Title of the Thesis PERSONALIZED DYNAMIC PRICING OF PERISHABLE RETAIL PRODUCTS USING A CONTEXTUAL BANDIT-BASED LEARNING FRAMEWORK INCORPORATING CUSTOMER SEGMENTATION AND INVENTORY MANAGEMENT

Total Pages: 66 Name of the scholar: Aatif Ameer

Supervisor(s): Dr. S. K. Garg

Department: Department of Mechanical Engineering

This is the report that the above thesis was scanned for similarity detection. The process and outcome are given below:

Software used: Turnitin Similarity Index: 6% Total Word Count: 10911

Date:

Candidate's Signature **Signature of Supervisor(s)**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

CERTIFICATE OF FINAL THESIS SUBMISSION

(1) Name: **Aatif Ameer**
(2) Roll No.: **23/IEM/02**
(3) Thesis Title: **"Personalized Dynamic Pricing Of Perishable Retail Products Using A Contextual Bandit-Based Learning Framework Incorporating Customer Segmentation and Inventory Management"**
(4) Degree for which the thesis is submitted: **M.Tech.**
(5) Faculty of the University to which the thesis is submitted: **Prof. S.K Garg**
(6) Thesis Preparation Guide was referred to for preparing the thesis: **YES**
(7) Specifications regarding thesis format have been closely followed: **YES**
(8) The contents of the thesis have been organized based on the guidelines: **YES**
(9) The thesis has been prepared without resorting to plagiarism: **YES**
(10) All sources used have been cited appropriately: **YES**
(11) The thesis has not been submitted elsewhere for a degree: **YES**
(12) Submitted 2 spiral bound copies plus one CD: **YES**

Signature of Supervisor

Dr. S.K. Garg

Signature of Candidate

Aatif Ameer

23/IEM/02

DELHI TECHNOLOGICAL UNIVERSITY
Shahbad Daulatpur, Main Bawana Road, Delhi-42

Proforma for Submission of M.Tech. Major Project

01. Name of the Student...AATIF AMEER.......

02. Enrolment No ..23/IEM/02............

03. Year of Admission ..2023.......................

04. Programme M.Tech., Branch..2023-25....

05. Name of Department...Mechanical Engineering

06. Admission Category i.e. Full Time/ Full Time (Sponsored)/ Part Time:...Full time....

07. Applied as Regular/ Ex-student.....Regular........

08. Span Period Expired on ....

09. Extension of Span Period Granted or Not Granted (if applicable)...............................

10.Title of Thesis/Major Project...personalized dynamic pricing of perishable Retail products using Contextual Bandit-Based Learning framework

11. Name of Supervisor....Dr. S.K. Garg...

12. Result Details (Enclose Copy of Mark sheets of all semesters) :

S. No.	Semester	Passing Year	Roll No.	Marks Obtained	Max. Marks	% of Marks	Details of Back Paper Cleared (if any)
01	1st	2024	23/IEM/02			9.176	
02	2nd	2024	23/IEM/02			8.65	
03	3rd	2025	23/IEM/02			8.333	
04	4th (P/T only)						
05	5th (P/T only)						

13. Fee Details (Enclose the Fee Receipt):

Amount Paid (in Rs.)	Receipt No.	Date
3000/-..............	..DU01232175..	28...../..05.../..25

Mohd Aatif
Signature of Student

It is certified that the name of Examiners for evaluation of the above thesis/ project have already been recommended by the BOS.

Signature of Supervisor

Signature of HOD with Seal

(Instructions for filling up the Form may see on back side please.)

e-Receipt for State Bank Collect Payment

REGISTRAR, DTU (RECEIPT A/C)

BAWANA ROAD, SHAHABAD DAULATPUR, , DELHI-110042
Date: 28-May-2025

SBCollect Reference Number :	DUO1232175
Category :	Miscellaneous Fees from students
Amount :	₹3000
University Roll No :	23/IEM/02
Name of the student :	Aatif Ameer
Academic Year :	2023-25
Branch Course :	Industrial engineering and management
Type/Name of fee :	Others if any
Remarks if any :	M.Tech Thesis Submission
Mobile No. of the student :	8700492973
Fee Amount :	3000
Transaction charge :	0.00
Total Amount (In Figures) :	3,000.00
Total Amount (In words) :	Rupees Three Thousand Only
Remarks :	

Notification 1:	Late Registration Fee, Hostel Room rent for internship, Hostel cooler rent, Transcript fee (Within 5 years Rs.1500/- & $150 in USD, More than 5 years but less than 10 years Rs.2500/- & $250 in USD, More than 10 years Rs.5000/- & $500 in USD) Additional copies Rs.200/- each & $20 in USD each, I-card fee,Character certificate Rs.500/-.
Notification 2:	Migration Certificate Rs.500/-, Bonafide certificate Rs.200/-, Special certificate (any other certificate not covered in above list) Rs.1000/-,Provisional certificate Rs.500/-, Duplicate Mark sheet (Within 5 years Rs.2500/- & $250 in USD, More than 5 years but less than 10 years Rs.4000/- & $400 in USD, More than 10 years Rs.10000/- & $1000 in USD)

Thank you for choosing SB Collect. If you have any query / grievances regarding the transaction, please contact us

Toll-free helpline number i.e. 1800-1111-09 / 1800 - 1234/1800 2100

Email -: sbcollect@sbi.co.in

Print Close

Delhi Technological University

(Formerly Delhi College of Engineering)

THE RESULT OF THE CANDIDATE WHO APPEARED IN THE FOLLOWING EXAMINATION HELD IN NOV 2023 IS DECLARED AS UNDER:-

Master of Technology(Industrial Engineering and Management), I-SEMESTER

Result Declaration Date : 04-03-2024 **Notification No: 1660**

IEM501 : Data Analytics IEM503 : Production & Operation Management IEM5205 : Principles of Managment IEM5305 : Total Quality Management IEM5407 : Product Design & Development

Sr.No	Roll No.	Name of Student	IEM501	IEM503	IEM5205	IEM5305	IEM5407	SGPA	TC	CGPA	Failed Courses
			4.00	4.00	2.00	3.00	4.00				
1	23/IEM/01	RAVI RANJAN	F	F	A+	O	A+	6.46	9	---	IEM501
2	23/IEM/02	AATIF AMEER	O	B+	A+	O	O	9.18	17	9.176	
3	23/IEM/03	MAHESH SAROHA	A+	A	A	A	O	8.71	17	8.706	
4	23/IEM/04	REDDI DUSHYANTH VENKATA SAI KRISHNA	A+	B+	A+	A	A	8.12	17	8.118	
5	23/IEM/05	DIVYANSH	C	C	A	C	B+	5.82	17	5.824	
6	23/IEM/06	RAJENDER	A+	B	A	A	A	7.76	17	7.765	
7	23/IEM/07	PIYUSH KUMAR	A+	B	A+	B	A	7.53	17	7.529	
8	23/IEM/08	ISHAN KOTNALA	C	F	B	C	B	4.18	13	---	IEM503
9	23/IEM/09	LOKESH KUMAR	A+	B+	A+	B+	A	7.94	17	7.941	
10	23/IEM/10	DHRUV SHANKAR SAXENA	A+	A	O	O	A+	9.06	17	9.059	
11	23/IEM/11	SHISHIR	A+	A+	A+	A+	A+	9	17	9	
12	23/IEM/12	MORIE MEYER KOUNA FERRAND	C	P	B+	B	B+	5.65	17	5.647	
13	23/IEM/13	FREDRICK KABWE	C	B	A	B+	B+	6.41	17	6.412	

IEM501 : Data Analytics IEM503 : Production & Operation Management

Sr.No	Roll No.	Name of Student	IEM501	IEM503	SGPA	TC	CGPA	Failed Courses
			4.00	4.00				
14	23/IEM/501	PRAMOD	C	F	2.5	4	---	IEM503

OIC (Results)

Controller of Examination

Note:Any discrepancy in the result in r/o name/roll no/registration/marks/grades/course code/title should be brought to the notice of Controller of Examination/OIC(Results) within 15 days of declaration of result in the prescribed proforma.

Delhi Technological University

(Formerly Delhi College of Engineering)

THE RESULT OF THE CANDIDATE WHO APPEARED IN THE FOLLOWING EXAMINATION HELD IN MAY 2024 IS DECLARED AS UNDER:-

Master of Technology(Industrial Engineering and Management), II-SEMESTER

Result Declaration Date : 16-07-2024

Notification No: 1691

IEM502 : OPERATIONS RESEARCH

Sr.No	Roll No.	Name of Student	IEM502	SGPA	TC	Failed Courses
			4.00			
1	23/IEM/501	PRAMOD	C	5	4	

IEM502 : OPERATIONS RESEARCH IEM504 : SUPPLY CHAIN MANAGEMENT IEM5210 : Contemporary Issues in Industrial Engineering and Managment IEM5304 : International Logistics and Warehouse Managment IEM5404 : INDUSTRY 4.0 & SMART MANUFACTURING

Sr.No	Roll No.	Name of Student	IEM502	IEM504	IEM5210	IEM5304	IEM5404	SGPA	TC	Failed Courses
			4.00	4.00	2.00	3.00	4.00			
2	23/IEM/01	RAVI RANJAN	O	B+	A+	A	O	8.82	17	
3	23/IEM/02	AATIF AMEER	A+	B+	O	A+	A+	8.65	17	
4	23/IEM/03	MAHESH SAROHA	O	A	A	A+	A+	8.88	17	
5	23/IEM/04	REDDI DUSHYANTH VENKATA SAI KRISHNA	A+	B	A	A	A+	8	17	
6	23/IEM/05	DIVYANSH	B+	P	B	B+	A+	6.65	17	
7	23/IEM/06	RAJENDER	A+	B	A	A	O	8.24	17	
8	23/IEM/07	PIYUSH KUMAR	A	B	A	A	O	8	17	
9	23/IEM/08	ISHAN KOTNALA	P	C	B+	B+	A	6.06	17	
10	23/IEM/09	LOKESH KUMAR	A+	A	A+	A+	A+	8.76	17	
11	23/IEM/10	DHRUV SHANKAR SAXENA	A+	A	O	O	O	9.29	17	
12	23/IEM/11	SHISHIR ACHARYA	A+	A	A	A+	O	8.88	17	
13	23/IEM/12	MORIE MEYER KOUNA FERRAND	A	C	A	A	A	7.29	17	
14	23/IEM/13	FREDRICK KABWE	A	B	A	A	A	7.53	17	

OIC (Results)

Controller of Examination

Note: Any discrepancy in the result in r/o name/roll no/registration/marks/grades/course code/title should be brought to the notice of Controller of Examination/OIC(Results) within 15 days of declaration of result in the prescribed proforma.

Delhi Technological University

(Formerly Delhi College of Engineering)

THE RESULT OF THE CANDIDATE WHO APPEARED IN THE FOLLOWING EXAMINATION HELD IN NOV 2024 IS DECLARED AS UNDER:-

Master of Technology(Industrial Engineering and Management), III-SEMESTER

Result Declaration Date : 12-03-2025

Notification No: 1798

IEM5205 : Principles of Managment IEM5305 : Total Quality Management IEM5407 : Product Design & Development

Sr.No	Roll No.	Name of Student	IEM5205	IEM5305	IEM5407	SGPA	TC	Failed Courses
			2.00	3.00	4.00			
1	23/IEM/501	PRAMOD	B+	C	B+	6.333	9	

IEM601 : MAJOR PROJECT I IEM6201 : E- Commerce IEM6305 : GLOBAL BUSINESS MANAGEMENT IEM6405 : Advanced Operation Research

Sr.No	Roll No.	Name of Student	IEM601	IEM6201	IEM6305	IEM6405	SGPA	TC	Failed Courses
			3.00	2.00	3.00	4.00			
2	23/IEM/01	RAVI RANJAN	A+	A+	A+	O	9.333	12	
3	23/IEM/02	AATIF AMEER	A+	A+	A+	B+	8.333	12	
4	23/IEM/03	MAHESH SAROHA	A	O	A+	O	9.250	12	
5	23/IEM/04	REDDI DUSHYANTH VENKATA SAI KRISHNA	A+	A+	A+	A+	9.000	12	
6	23/IEM/05	DIVYANSH	A+	A	B+	B	7.333	12	
7	23/IEM/06	RAJENDER	O	A+	A	A+	9.000	12	
8	23/IEM/07	PIYUSH KUMAR	A+	A+	B+	B+	7.833	12	
9	23/IEM/08	ISHAN KOTNALA	A+	A+	A+	B+	8.333	12	
10	23/IEM/09	LOKESH KUMAR	A+	A+	A+	B+	8.333	12	
11	23/IEM/10	DHRUV SHANKAR SAXENA	O	O	O	O	10.000	12	
12	23/IEM/11	SHISHIR ACHARYA	O	O	O	O	10.000	12	
13	23/IEM/12	MORIE MEYER KOUNA FERRAND	A+	A	A+	A	8.500	12	
14	23/IEM/13	FREDRICK KABWE	A+	A+	A+	A+	9.000	12	

OIC (Results)

Controller of Examination

Note: Any discrepancy in the result in r/o name/roll no/registration/marks/grades/course code/title should be brought to the notice of Controller of Examination/OIC(Results) within 15 days of declaration of result, in the prescribed proforma.

AATIF AMEER | +91-8700492973 | aatifameer_23iem02@dtu.ac.in

EDUCATION

Degree/Certificate	Institute	CGPA / %	Year
MTech, Industrial and Management Engineering	DELHI TECHNOLOGICAL UNIVERSITY, DELHI	8.319/10	2023-25
BTech, Mechanical Engineering	G L BAJAJ INSTITUTE OF TECHNOLOGY AND MANAGEMENT	8.01/10	2017-21
Senior Secondary School	RADIANT ACADEMY, NOIDA	73/100	2016-17
Secondary School	RADIANT ACADEMY, NOIDA	7.8/10	2014-15

Self Projects

Customer Churn Prediction

Approach	• Collected and pre-processed data from the company's CRM, focusing on customer demographics, service usage patterns, and historical churn behaviour. • Applied exploratory data analysis (EDA) to uncover trends and correlations. • Developed a predictive model using Random Forest and Logistic Regression, fine-tuned with cross-validation and hyperparameter optimization.
Result	• Achieved an accuracy of 82% with the model and reduced customer churn by 15% through proactive retention strategies informed by model predictions.

Sales Forecasting for Retail

Objective	• To build a sales forecasting model to help a retail company optimize inventory management and improve supply chain efficiency.
Approach	• Gathered historical sales data, promotional information, and seasonal trends for multiple product categories. • Performed time series analysis using ARIMA and Prophet models to capture seasonality and trend patterns. • Conducted feature engineering to account for external factors such as promotions, holidays, and economic conditions
Result	• Achieved a mean absolute percentage error (MAPE) of 7%, enabling more accurate sales predictions and a 10% reduction in overstock and understock scenarios.

Sentiment Analysis of Product Reviews

Objective	• To analyze customer sentiment from product reviews and categorize them as positive, neutral, or negative, helping the company improve product quality and customer satisfaction.
Approach	• Extracted and cleaned product review data from online platforms using web scraping. • Applied natural language processing (NLP) techniques like tokenization, TF-IDF, and sentiment analysis using a pre-trained VADER model. • Visualized results with Power BI to provide actionable insights, such as the most common customer pain points.
Result	• Delivered a dashboard with sentiment insights, revealing that 65% of the reviews were positive, leading to targeted improvements in the product features most criticized by customers.

COURSEWORK & SKILLS

- **COURSEWORK|** Statistical Modelling for Business Analytics | Applied Machine Learning| Probability & Statistics
- **SKILLS |** SQL **|** Python| Excel| ML Libraries: NumPy, Pandas, Matplotlib, Seaborn, Scikit-learn | Data analysis and Data visualization | Power BI |Machine Learning |
- **SOFT SKILLS**| Decision Making | Adaptability | Problem Solving | Analytical Thinking | Leadership | Teamwork
- **CERTIFICATIONS|** Basics of Machine Learning |SQL for Data Science| Basics of Python

ACHIEVEMENTS & EXTRACURRICULAR

- Rotaract club- Public relations DTU Delhi (2023-Present).
- Qualified GATE XE RANK 1940
- Class representative in B.TECH

APC Payment Confirmation.

1 message

International Journal of Environmental Sciences . <accounts@editorsijes.net> Mon, Jun 16, 2025 at 6:51 AM
To: aatifameer1998@gmail.com

Dear Aatif Ameer,

Greetings from the International Journal of Environmental Sciences.

We are writing to confirm that we have successfully received your payment towards the Article Processing Charges (APC) for your accepted manuscript entitled *"Personalized Dynamic Pricing Strategies for Perishable Products: A Contextual Bandit Algorithm Approach"* (Manuscript ID: IJES - 114).

Thank you for completing the payment process. Please note that the official invoice for the APC will be issued and sent to you via email within 24 hours.

Should you require any further assistance or have specific invoice requirements, feel free to contact us.

We appreciate your contribution and thank you for choosing the International Journal of Environmental Sciences to publish your research.

Warm regards,
Editorial Office

International Journal of Environmental Sciences

Acceptance Notification- IJES

1 message

International Journal of Environmental Sciences <ijes.editor@theaspd.com> Wed, Jun 11, 2025 at 3:53 PM
To: aatifameer1998@gmail.com

Dear Author(s),

Greetings of the day!

We hope this email finds you in great spirits.

We are writing to inform you that your article   has been accepted for publication in the upcoming issue of the **International Journal of Environmental Sciences, ISSN: 2229-7359.**

If you have any questions or need additional information, please feel free to contact us at your convenience.

Regards

Editorial Team,
International Journal of Environmental Sciences
Website: https://www.theaspd.com/ijes.php
ISSN: 2229-7359

Acceptance Letter IJES - 114.pdf
402 KB

International Journal of Environmental Sciences,
ISSN: 2229-7359

Date: 11-06-2025

Article ID:- IJES - 114

Dear Author(s)

Aatif Ameer
(Delhi Technological University)

We would like to inform you that your manuscript has been accepted for publication in International Journal of Environmental Sciences ISSN: 2229-7359

Manuscript Title: **Personalized Dynamic Pricing Strategies for Perishable Products: A Contextual Bandit Algorithm Approach**

After receiving the publication fee, we will email you the galley proof for your final confirmation.

Thanks for submission of your work with us.

Regards,

Mihai V. Putz
Editor in Chief
International Journal of Environmental Sciences
ISSN:2229-7359
https://theaspd.com/index.php/ijes/index

ELSEVIER
Scopus

OPEN ACCESS
This journal also publishes Open Access articles

Manuscript IJES - 114: "Personalized Dynamic Pricing Strategies for Perishable Products: A Contextual Bandit Algorithm Approach" - Under Review

1 message

Turkish Online Journal of Qualitative Inquiry <editor@tojqi.net> Wed, Jun 11, 2025 at 3:19 PM
To: aatifameer1998@gmail.com

Dear Author(s),

Thank you for submitting your manuscript, "**Personalized Dynamic Pricing Strategies for Perishable Products: A Contextual Bandit Algorithm Approach**" to the journal. Your article, identified as **IJES - 114**, has been successfully received and is currently undergoing peer review.

We will notify you of the final decision regarding your manuscript shortly. Upon acceptance, an official acceptance letter and the corresponding invoice for publication charges will be issued promptly.

Warm regards.